МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
МЕЖДУНАРОДНАЯ АКАДЕМИЯ ИНФОРМАТИЗАЦИИ
СОЮЗ МАШИНОСТРОИТЕЛЕЙ РОССИЙСКОЙ ФЕДЕРАЦИИ
МИНИСТЕРСТВО ПРОМЫШЛЕННОСТИ, ИННОВАЦИОННЫХ
И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ РЯЗАНСКОЙ ОБЛАСТИ
РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ В.Ф. УТКИНА

СОВРЕМЕННЫЕ ТЕХНОЛОГИИ В НАУКЕ И ОБРАЗОВАНИИ

СТНО-2023

VI МЕЖДУНАРОДНЫЙ НАУЧНО-ТЕХНИЧЕСКИЙ ФОРУМ

Сборник трудов

Том 3

Рязань
2023

УДК 004 + 001.1 + 681.2+ 681.2+ 681.3+681.5
С 568

Современные технологии в науке и образовании – СТНО-2023 [текст]: сб. тр. VI междунар. науч.-техн. форума: в 10 т. Т.3./ под общ. ред. О.В. Миловзорова. – Рязань: Рязан. гос. радиотехн. ун-т, 2023.

ISBN 978-5-7722-0373-6

Т.3: – 196 с.,: ил.

ISBN 978-5-7722-0376-7

Сборник включает труды участников VI Международного научно-технического форума «Современные технологии в науке и образовании» СТНО-2023.

В сборнике освещаются вопросы математического моделирования, новых технологий в радиотехнике, телекоммуникациях, электротехнике и радиоэлектронике, вопросы полупроводниковой наноэлектроники, приборостроения, лазерной, микроволновой техники, силовой промышленной электроники, новые технологии в измерительной технике и системах, биомедицинских системах, алгоритмическое и программное обеспечение вычислительной техники, вычислительных сетей и комплексов, вопросы систем автоматизированного проектирования, обработки изображений и управления в технических системах, перспективные технологии в машиностроительном и нефтехимическом производствах, новые технологии и методики в высшем образовании, в т.ч. вопросы гуманитарной и физико-математической подготовки студентов, обучения их иностранным языкам, перспективные технологии электронного обучения, в том числе, дистанционного, вопросы экономики, управления предприятими и персоналом, менеджмента, а также вопросы гуманитарной сферы.

Авторская позиция и стилистические особенности сохранены.

УДК 004 + 001.1 + 681.2+ 681.2+ 681.3+681.5

ISBN 978-5-7722-0373-6

ISBN 978-5-7722-0376-7

ИНФОРМАЦИЯ О VI МЕЖДУНАРОДНОМ ФОРУМЕ «СОВРЕМЕННЫЕ ТЕХНОЛОГИИ В НАУКЕ И ОБРАЗОВАНИИ» СТНО-2023

VI Международный научно-технический форум «Современные технологии в науке и образовании» СТНО-2023 состоялся 01.03.2022-03.03.2022 в г. Рязань в Рязанском государственном радиотехническом университете имени В.Ф. Уткина.

В рамках форума «Современные технологии в науке и образовании» СТНО-2023 состоялась работа четырех Международных научно-технических конференций:

«Современные технологии в науке и образовании. Радиотехника и электроника», секции

- Радиотехнические системы и устройства;
- Телекоммуникационные системы и устройства;
- Цифровые информационные технологии реального времени;
- Промышленная силовая электроника, электроэнергетика и электроснабжение;
- Физика полупроводников, микро- и наноэлектроника;
- Микроволновая, оптическая и квантовая электроника;
- Актуальные задачи химических технологий;

«Современные технологии в науке и образовании. Вычислительная техника и автоматизированные системы», секции

- Алгоритмическое и программное обеспечение вычислительных систем и сетей;
- ЭВМ и системы;
- Информационные технологии в конструировании электронных средств;
- Модели искусственного интеллекта в САПР;
- Информационные системы и защита информации;
- Математические методы в научных исследованиях;
- Обработка данных, изображений и управление в технических системах;
- Геоинформационные и космические технологии;
- Автоматизация производственно-технологических процессов в приборо- и машиностроении;
- Информационно-измерительные устройства и системы в технике и медицине.
- Стандартизация и управление качеством;
- Информационные системы и технологии;

«Современные технологии в науке и образовании. Экономика и управление», секции;

- Современные технологии государственного и муниципального управления;
- Экономика, менеджмент и организация производства;
- Бухгалтерский учет, анализ и аудит;
- Экономическая безопасность;
- Внешнеэкономическая деятельность;

«Современные технологии в науке и образовании. Новые технологии и методы в высшем образовании», секции

- Современные технологии электронного обучения;
- Иностранный язык в техническом вузе;
- Лингвистика и межкультурная коммуникация;
- Направления и формы гуманитаризации высшего образования и гуманитарная подготовка студентов;
- Методы преподавания и организация учебного процесса в вузе;
- Физико-математическая подготовка студентов;
- Особенности военного образования на современном этапе.

Организационный комитет Форума:

Чиркин М.В., ректор, д.ф.-м.н., проф. – председатель

Гусев С.И., проректор по научной работе и инновациям, д.т.н., проф. – зам. председателя;

Корячко А.В., проректор по учебной работе, к.т.н., доц. – зам. председателя;

Миловзоров О.В., зам. начальника управления организации научных исследований, к.т.н, доц. – координатор, главный редактор сборника трудов Форума;

Миронов В.В., ответственный редактор сборника трудов Форума;

Устинова Л.С., начальник отдела информационного обеспечения – отв. за информационную поддержку;

Трубицына С.Г., вед. инженер – секретарь оргкомитета;

Благодарова И.А., ведущий программист – секретарь оргкомитета;

члены оргкомитета:

Бабаян П.В., зав. кафедрой автоматики и информационных технологий в управлении, к.т.н., доц.;

Бухенский К.В., зав. кафедрой высшей математики, к.ф.-м.н., доц.;

Витязев В.В., зав. кафедрой телекоммуникаций и основ радиотехники, д.т.н., проф.;

Губарев А.В., доцент кафедры информационно-измерительной и биомедицинской техники, к.т.н., доц.;

Дмитриев В.Т., зав. кафедрой радиоуправления и связи, к.т.н., доц.;

Евдокимова Е.Н., зав. кафедрой экономики, менеджмента и организации производства, д.э.н., проф.;

Еремеев В.В., директор НИИ «Фотон», д.т.н., проф.;

Есенина Н.Е., зав. кафедрой иностранных языков, к.п.н., доц.;

Жулев В.И., зав. кафедрой информационно-измерительной и биомедицинской техники, д.т.н., проф.;

Карпунина Е.В., доцент кафедры экономической безопасности, анализа и учета, к.э.н., доц.;

Клейносова Н.П., директор центра дистанционного обучения, к.п.н., доц.;

Клочко В.К., профессор кафедры автоматики и информационных технологий в управлении, д.т.н., проф.;

Коваленко В.В., зав. кафедрой химической технологии, к.т.н., доц.;

Корячко В.П., зав. кафедрой систем автоматизированного проектирования вычислительных средств, д.т.н., проф.;

Костров Б.В., зав. кафедрой электронных вычислительных машин, д.т.н., проф.;

Кошелев В.И., зав. кафедрой радиотехнических систем, д.т.н., проф.;

Крошилина С.В., доцент кафедры вычислительной и прикладной математики, к.т.н., доц.;

Круглов С.А., зав. кафедрой промышленной электроники, д.т.н., доц.;

Куприна О.Г., доцент кафедры иностранных языков, к.филол.н., доц.;

Ленков М.В., зав. кафедрой автоматизации информационных и технологических процессов, к.т.н., доц.

Литвинов В.Г., зав. кафедрой микро- и наноэлектроники, д.ф.-м.н., доц.;

Лукьянова Г.С., доцент кафедры высшей математики, к.ф.-м.н., доц.;

Мельник О.В., профессор кафедры информационно-измерительной и биомедицинской техники, д.т.н., доц.;

Митрошин А.А., доцент кафедры систем автоматизированного проектирования вычислительных средств, к.т.н., доц.;

Овечкин Г.В., зав. кафедрой вычислительной и прикладной математики, д.т.н., проф.;

Паршин Ю.Н., зав. кафедрой радиотехнических устройств, д.т.н., проф.;

Перфильев С.В., зав. кафедрой государственного, муниципального и корпоративного управления, д.э.н., проф.;

Подгорнова Н.А., доцент кафедры государственного, муниципального и корпоративного управления, к.э.н., доц.;

Пржегорлинский В.Н., зав. кафедрой информационной безопасности, к.т.н., доц.;

Пылькин А.Н., профессор кафедры вычислительной и прикладной математики, д.т.н., проф.;

Рохлина Т.А., доцент кафедры иностранных языков, к.филол.н., доц.;

Сапрыкин А.Н., доцент кафедры систем автоматизированного проектирования вычислительных средств, к.т.н., доц.;

Семенов А.Р., ст.преподаватель кафедры химической технологии, к.ф.-м.н.;

Серебряков А.Е., зам. зав. кафедрой электронных приборов, к.т.н.;

Скрипкина О.В. доцент кафедры экономической безопасности, анализа и учета, к.э.н., доц.;

Соколов А.С., зав. кафедрой истории, философии и права, д.и.н;

Таганов А.И., профессор кафедры космических технологий, д.т.н., проф.;

Тарасова В.Ю., ассистент кафедры электронных вычислительных машин, магистр;

Харитонов А.Ю., нач. военного учебного центра, полковник, к.т.н., доц;

Холопов С.И., декан ф-та автоматики и информационных технологий в управлении, зав. кафедрой автоматизированных систем управления, к.т.н., доц.;

Чеглакова С.Г., зав. кафедрой экономической безопасности, анализа и учета, д.э.н., проф.;

Челебаев С.В., доцент кафедры автоматизированных систем управления, к.т.н., доц.

МЕЖДУНАРОДНАЯ НАУЧНО-ТЕХНИЧЕСКАЯ КОНФЕРЕНЦИЯ «СОВРЕМЕННЫЕ ТЕХНОЛОГИИ В НАУКЕ И ОБРАЗОВАНИИ. РАДИОТЕХНИКА И ЭЛЕКТРОНИКА»

СЕКЦИЯ «ЦИФРОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ РЕАЛЬНОГО ВРЕМЕНИ»

УДК 656.025:004.942; ГРНТИ 47.47

ПЕРСПЕКТИВЫ ИСПОЛЬЗОВАНИЯ ЦИФРОВЫХ ТЕХНОЛОГИЙ НА ТРАНСПОРТЕ

Е.В. Громак
Белорусский национальный технический университет,
Республика Беларусь, Минск, mrloloche@gmail.com

Аннотация. В работе рассматривается возможность использования цифровых технологий в транспортной отрасли. Приводятся основные преимущества и недостатки, а также отрасли транспорта, в которых они применимы.
Ключевые слова: информационные технологии, транспорт, логистика, развитие.

USE OF DIGITAL TECHNOLOGIES IN THE TRANSPORT INDUSTRY

E.V.Gromak
Belarusian National Technical University,
Belarus, Minsk, mrloloche@gmail.com

The summary. The paper considers the possibility of using digital technologies in the transport industry. The main advantages and disadvantages are given, as well as the transport sectors in which they are applicable.
Keywords: information technology, transport, logistics, development.

В результате пандемии, в логистике повысились темпы развития электронной торговли, ускоряется сквозная цифровизация цепочек поставок и доставки, в том числе с помощью беспилотников. Также растет спрос на транзакции и обработку грузов через цифровые платформы на основе технологии распределенных книг, включая блокчейн, отслеживание грузов через технологию Интернет вещей, управление запасами с помощью аналитики, а также планирование ремонта и технического обслуживания с помощью предиктивной аналитики.

В списке самых востребованных технологий — беспроводная связь, искусственный интеллект, в том числе компьютерное зрение, речевые технологии, системы поддержки принятия решений, системы распределенного реестра, решения в области виртуальной и дополненной реальности. Спрос транспорта и логистики на цифровые технологии в 2021 г. составил ₽94,4 млрд, а к 2030 г. может вырасти до ₽626,6 млрд. [1].

Интернет вещей (IoT)

Интернет вещей (Internet of Things, IoT) — это сеть взаимодействующих между собой объектов, имеющих подключение к интернету и обменивающихся данными. Концепция IoT можетз начительно улучшить многие сферы нашей жизни и помочь создать более удобный, умный и безопасный мир.

В 2019 году первое место по применению интернет вещей заняла промышленность, второе место – транспорт. Технологии IoT на транспорте — это сеть устройств, которые могут собирать и передавать данные об автобусах, трамваях и автомобилях через интернет, используя IP-камеры, датчики и сенсоры для автоматического контроля всего, что касается транспортного средства, от его местоположения до мониторинга давления в шинах. На осно-

ве этих данных разрабатываются такие услуги, как карты пробок, табло прибытия общественного транспорта и приложения для парковки.История развития IoT показана на рисунке 1.

Рис. 1. История развития Интернет вещей (IoT)

В России драйвером развития IoT в транспортном секторе является Москва, где местные власти мотивируют участников рынка к цифровизаци и отрасли: с 2012 по 2021 год на интеллектуальную транспортную систему в Москве было выделено 138 млрд рублей. Вслед за столицей аналогичные решения внедряются и в регионах.

IoT-устройства помогают создавать цифровых "двойников" автобусов, трамваев, метро и других видов транспорта. Это модель, содержащая информацию обо всех механизмах, их техническом состоянии играфикахремонта. Данные постоянно обновляются. Системапозволяетперейти от планового ремонтак ремонту потребованию, экономя запасные части и снижая аварийность [2].

Искусственный интеллект (AI)

Решения на основе ИИс окращают время и общие затраты на складирование и дистрибуцию благодаря анализу существующих маршрутов, выявлению узких мест и концентрации налучших маршрутах. Инструменты обработки данных на основе ИИ предоставляют подробную информацию о движении товаров в режиме реального времени и помогают правильно прогнозировать сроки доставки. Главное преимущество ИИ заключается в том, что его можно использовать один раз и до тех пор, пока он необходим.

Задача таких технологий - помочь людям принимать правильные решения,предупредить их об опасности и сделать жизнь проще и быстрее.

Например,если водитель автобуса или троллейбуса начнет засыпать,электронное устройство сможет подать ему сигнал.Оно также позволит людям оплачивать проезд в общественном транспорте с помощью биометрического распознавания лица, без кондуктора, транспортной карты или смартфона.

Сейчас в России испытывают и начинают внедрять систему контроля усталости водителя. Это особенно важно для общественного транспорта (автобусов, троллейбусов, трамваев, поездов метро). Есть разные причины, по которым водитель может уснуть за рулем и спровоцировать аварию. Но с новой системой такого не произойдет.

Специальная камера фиксирует действия водителя, например частоту моргания, движений, мимики лица. Исходя из этих данных электроника определяет степень усталости че-

ловека. Если водитель засыпает, отклоняется от маршрута система подает звуковой сигнал, чтобы привести водителя в чувства и разбудить его [3].

Программные роботы для автоматизации задач (RPA)

Рабочий процесс зачастую состоит из операций, выполняемых вручную и требующих больших временных затрат. В результате выполнения этих операций снижается производительность и мотивация сотрудников.

Любые шаблонные задачи, для которых может составлен алгоритм действий, могут быть выполнения программным роботом. Робот поэтапно записывает действия пользователя, для последующего воспроизведения.

Задача RPA заменить повторяющиеся задачи, выполняемые людьми, виртуальными роботами. Робот использует компьютер для запуска прикладных программ таким же образом, как и человек, работающий с этим программным обеспечением.

RPA - это автоматизация, осуществляющая взаимодействие с компьютером с использованием пользовательского интерфейса.

Например, при использовании данной технологии можно выполнять следующие виды работ:

1) инвентаризация;
2) обработка возвратов товара;
3) управление контрактами и счетами;
4) мониторинг серверов и приложений;
5) мониторинг большого количества транспортных единиц в разных системах;
6) работа диспетчерской службы и много другое.

Пример работы этой технологии на железнодорожном транспорте:

1) из информационной системы робото создает Excel файл в котором,согласно всем установленным маршрутам, содержатся номера вагонов;
2) переходя во вторую информационную систему, робот, следуя алгоритму, анализирует причины задержки в пути следования вагонов;
3) далее робот поочередно вводит номера вагонов и выгружает все найденные по ним акты в Excel файл. За сутки робот может проанализировать до 1200 вагонов [4].

Вследствие чего идет оптимизация штата сотрудников на 24 единицы.

На постоянно меняющимся рынке логистическим предприятиям недостаточно оптимизировать только уже существующие операции, необходимо также одновременно внедрять новые технологии, которые выходят далеко за рамки простых решений автоматизации. Компаниям приходится постоянно искать новые решения для того, чтобы они обеспечили им технологический прорыв и избавили от конкуренции.

Цифровые технологии позволяют значительно упростить процедуры внешней торговли засчет применения глобальных международных стандартов и моделей данных и создания правовой базы для трансграничной безбумажной торговли. Грамотная стратегия цифровой трансформации позволит повысить эффективность социально-экономического развития государства.

Библиографический список

1. CNEWS [Электронный ресурс]. Режим доступа: https://www.cnews.ru/reviews/it_v_transportnoj_otrasli_2021
2. Национальные проекты [Электронный ресурс]. Режим доступа: https://национальныепроекты.рф/news/kak-iskusstvennyy-intellekt-pomogaet-uluchshit-situatsiyu-na-transporte
3. Атрохов Н.А., Тураева О.П. Цифровизация грузовых перевозок. [Электронный ресурс]. – URL: www.esa-conference.ru.
4. Hightech [Электронный ресурс]. Режим доступа: https://hightech.fm/2020/07/15/iot-transport-russia.

УДК 621.396; ГРНТИ 47.47

РЕАЛИЗАЦИЯ ОБРАТНОГО ПОИСКА ЧЕНЯ

Н.С. Виноградов
Рязанский государственный радиотехнический университет имени В.Ф. Уткина,
Российская Федерация, Рязань, nikita-vinogradov11@rambler.ru

Аннотация. В работе рассматривается алгоритм поиска Ченя, используемый в БЧХ декодере. Рассматривается традиционный алгоритм поиска Ченя и описываются его проблемы. Предлагается реализация обратного поиска Ченя.
Ключевые слова: коды Боуза-Чоудхури-Хоквингема (БЧХ), программируемые логических интегральные схемы (ПЛИС), алгоритм поиска Ченя.

IMPLEMENTATION OF REVERCE CHIEN SEARCH

N.S. Vinogradov
Ryazan State Radio Engineering University named after V.F. Utkin,
Russia, Ryazan, nikita-vinogradov11@rambler.ru

The summary. The paper considers the Chen search algorithm used in the BCH decoder. The traditional Chen search algorithm is considered and its problems are described. The implementation of the reverse Chen search is proposed.
Keywords: Bose–Chaudhuri–Hocquenghem codes (BCH), field-programmable gate array (FPGA), Chien search algorithm.

Коды БЧХ активно используются в современных телекоммуникационных системах. Так, например, в стандартах DVB-S2 и DVB-S2X коды БЧХ используются в связке с LDPC кодами для помехоустойчивого кодирования [1]. В связи с высокими требованиями к пропускной способности в телекоммуникационных системах возникает необходимость в реализации кодеков БЧХ на ПЛИС. Кодирование с использованием БЧХ кодов выполняется по простому алгоритму деления полиномов, поэтому затраты ресурсов ПЛИС на реализацию БЧХ кодера достаточно малы. Декодер БЧХ, напротив, имеет сложный алгоритм, и при синтезе его на ПЛИС может затрачиваться большое количество ресурсов. В связи с этим возникает необходимость в разработке оптимального БЧХ декодера с требуемой пропускной способностью и с минимальными затратами ресурсов.

Схема БЧХ декодера из [1] приведена на рисунке 1.

Рис. 1. Схема БЧХ декодера

БЧХ декодер имеет три основных блока: блок расчета синдрома, блок решателя ключевого уравнения (РКУ) и блок поиска Ченя. Блок расчета синдрома принимает на вход кодовое слово и высчитывает синдром. Блок РКУ с помощью полученного синдрома рассчитывает полином локатора ошибок. Блок поиска Ченя используется для нахождения корней полинома локатора ошибок, которые указывают в каких битах кодового слова произошли ошибки. Помимо трех основных блоков также имеется блок исправления ошибок,

который принимает на вход результаты поиска Ченя и входное кодовое слово с FIFO, а на выход подает кодовое слово с исправленными ошибками.

Традиционный модуль поиска Ченя

Существует множество вариантов реализации поиска Ченя. Последовательный поиск Ченя, позволяющий за один такт проверять один корень, представлен в [1]. Однако, проверка одного корня за такт не позволяет получить высокую пропускную способность, поэтому чаще всего используют параллельную схему поиска Ченя, которая представлена в [2]. Традиционная схема параллельного поиска Ченя представлена на рисунке 2, где

- $\lambda_{1,\,2,\,\ldots,\,t}$ – коэффициенты полинома локатора ошибок,
- α – примитивный член поля Галуа,
- t – корректирующая способность BCH декодера,
- p – коэффициент распараллеливания поиска Ченя,
- i – номер итерации поиска Ченя.

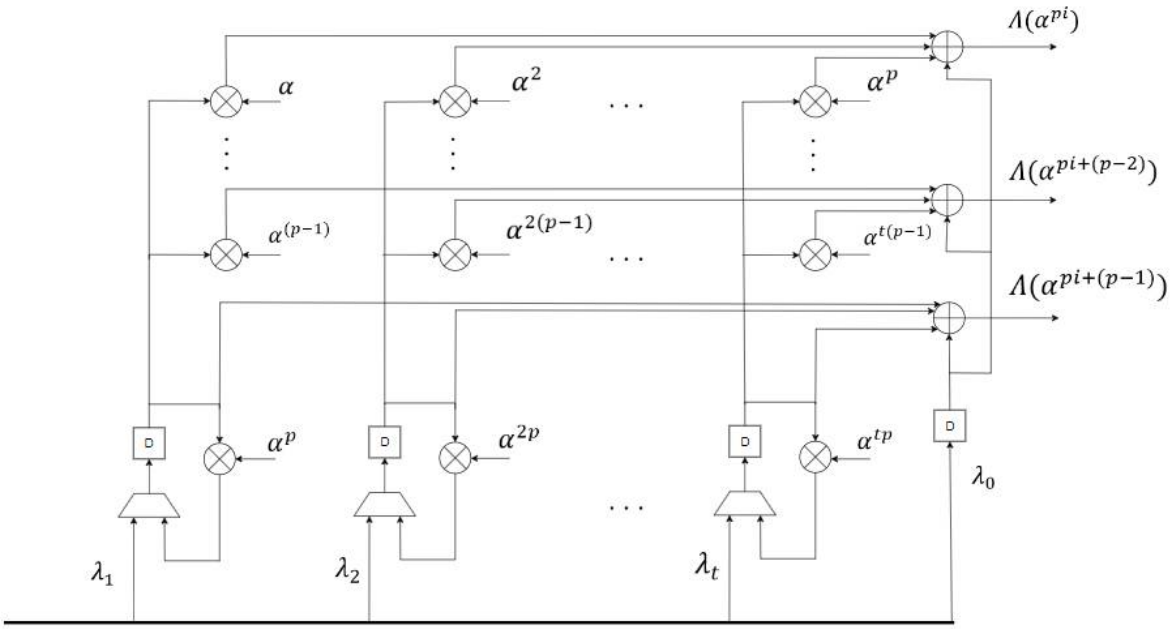


Рис. 2. Схема параллельного поиска Ченя

В качестве умножителей на рисунке 2 используются константные умножители поля Галуа. На первом такте на вход константных умножителей поля Галуа через мультиплексор поступают рассчитанные коэффициенты полинома локатора ошибок. На последующих тактах на вход константных умножителей подаются значения из регистров *D*.

Значения с выходов константных умножителей поля Галуа поступают на сумматоры поля Галуа, где происходит их сложение. После сложения всех t произведений, на выходах блоков сумматоров, получаются значения $\Lambda(\alpha^{pi})$, $\Lambda(\alpha^{pi+1})$, ..., $\Lambda(\alpha^{pi+(p-1)})$. Номер найденного корня можно выразить через номер ветви поиска Ченя, то есть $c \in (1, 2, ..., p)$, где c – номер ветви поиска Ченя. Количество ветвей поиска Ченя равно коэффициенту распараллеливания. Если $\Lambda(\alpha^{pi+c-1}) == 0$, то в $(pi+c-1)$-ом бите кодового слова зарегистрирована ошибка, которую необходимо исправить.

Учет укороченности БЧХ кодов

Максимальная длина кодового слова БЧХ кода равна:

$$N = 2^m - 1,$$

где N – максимальная длина кодового слова,

m – степень поля Галуа, используемого для построения БЧХ кода.

В современных системах связи часто используются укороченные БЧХ коды, длины которых меньше N. Так, например, в стандарте DVB-S2 и DVB-S2X используются только укороченные коды. Укороченные БЧХ коды получаются путем обнуления первых $(N - n)$ бит, где n – длина кодового слова для укороченного БЧХ кода. Схема поиска Ченя, рассмотренная выше для укороченных БЧХ кодов в начале проверяет $(N - n)$ корней, которые проверять нет смысла. Для ускорения работы поиска Ченя для укороченных БЧХ кодов применяют схему из [3], изображенную на рисунке 3.

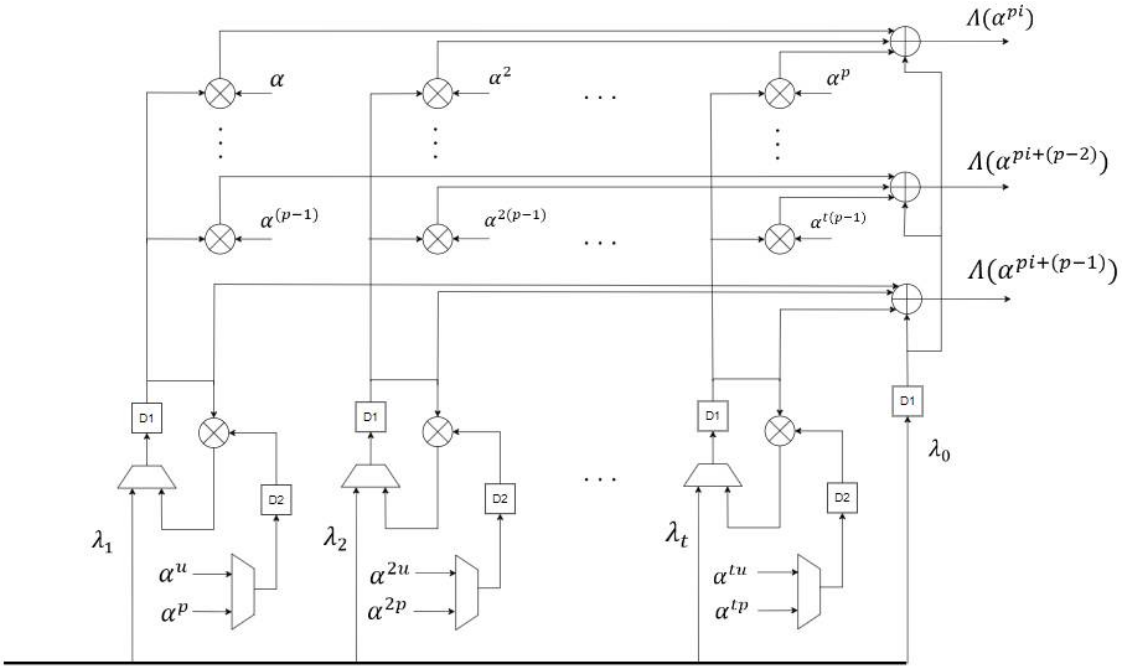


Рис. 3. Схема поиска Ченя с учетом укороченности БЧХ кода

Так, на вход регистра *D2*, относящегося к первому умножителю, через мультиплексор поступают константы α^p и α^u. Константа α^p используется в процессе поиска Ченя (аналогично с рисунком 2), а константа α^u используется для избавления от поиска корней полинома локатора ошибок в битах укороченного БЧХ кода, которые не используются, то есть принимаются равными 0. Значение u вычисляется по следующей формуле:

$$u = (2^m - 1 - n),$$

где m – степень поля Галуа,

n – размер кодового слова для укороченного БЧХ кода.

Обратный поиск Ченя

Процедура поиска Ченя проверяет значение полинома локатора ошибок при подстановке вместо x коэффициентов от α до $\alpha^{2\text{^}m-1}$. Такой порядок подстановки коэффициентов назовем прямым, а процедуру поиска Ченя соответственно – *прямым поиском Ченя*.

Одной из проблем *прямого поиска Ченя* для укороченных БЧХ кодов является необходимость хранения набора коэффициентов $(\alpha^u, \alpha^{2u}, ..., \alpha^{tu})$ для каждого режима работы БЧХ декодера (рисунок 3). В стандарте DVB-S2X количество режимов работы БЧХ декодера равно 49. Отказ от хранения таких наборов коэффициентов приведет к снижению скорости работы модуля поиска Ченя.

Введем понятие *обратного поиска Ченя*. Процедура *обратного поиска Ченя* характеризуется обратным порядком подстановки набора коэффициентов $(\alpha, \alpha^2, ..., \alpha^{2^{m-1}})$, то есть от $\alpha^{2^{m-1}}$ до α. Такой способ поиска Ченя позволяет получать значения полинома локатора ошибок начиная от последнего элемента поля Галуа. Соответственно, для укороченных БЧХ кодов это означает, что в модуле поиска Ченя не будет необходимости хранить наборы константных значений поля Галуа $(\alpha^u, \alpha^{2u}, ..., \alpha^{tu})$ для каждого режима работы БЧХ декодера. Таким образом, поиск Ченя будет начинаться от последнего элемента поля Галуа и заканчиваться, когда будут проверены n (длина кодового слова) значений полинома локатора ошибок.

Для реализации обратного поиска Ченя необходимо заменить все коэффициенты, используемые в схеме поиска Ченя, на обратные, как показано на рисунке 4.

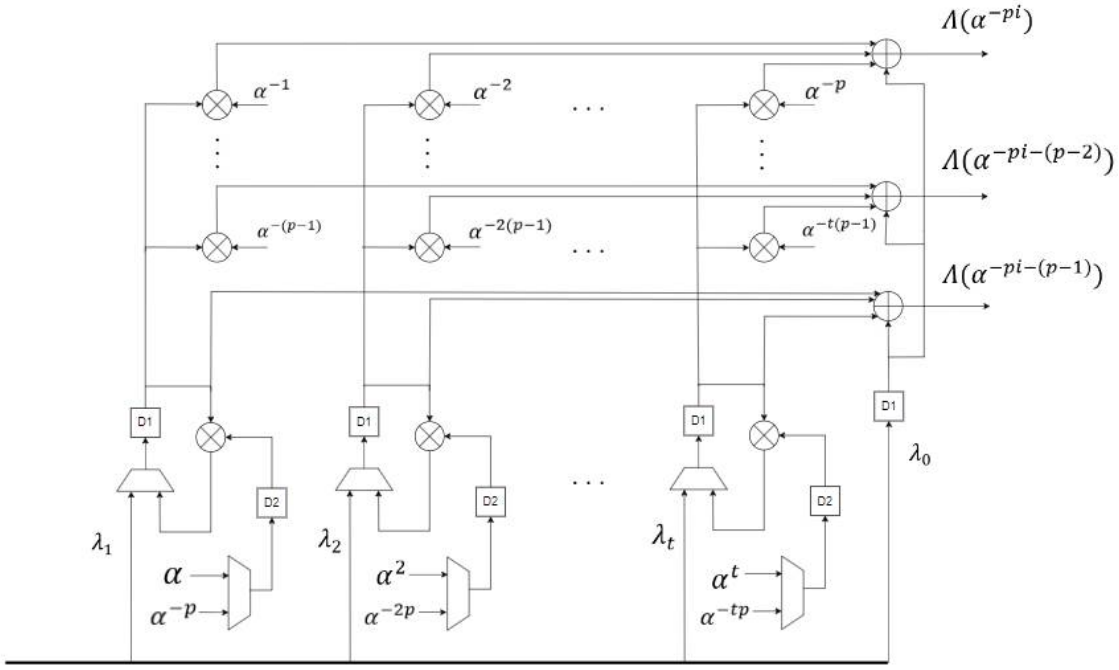


Рис. 4. Схема обратного поиска Ченя

В [4] дается определение обращения элемента поля. Получение обратного элемента поля Галуа α^{-i} может быть произведено по следующей формуле:

$$\alpha^{-i} = (\alpha^i)^{2^m - 2},$$

где α – примитивный элемент поля Галуа,

i – степень примитивного элемента поля Галуа,

m – степень поля Галуа GF(2^m).

Если просто заменить все константные значения поля Галуа на обратные, полноценного *обратного поиска Ченя*, который подразумевает, что в полином локатора ошибок вместо x подставляются значения от $\alpha^{2^{m-1}}$ до α, мы не получим.

Чтобы объяснить проблему, воспользуемся следующим свойством полей Галуа из [4]. Согласно малой теореме Ферма справедливо следующее равенство:

$$(\alpha^i)^{2^m-1} = 1.$$

Соответственно, получаем, что обратный поиск Ченя подразумевает подстановку в полином локатора ошибок вместо *x* набора коэффициентов *($\alpha^{2\text{^}m-1}$, $\alpha^{2\text{^}m-2}$, ... α)* или *(1, $\alpha^{2\text{^}m-2}$, ... α)*. Таким образом, первая ветвь поиска Ченя на первом цикле поиска Ченя должна умножить коэффициенты полинома локатора ошибок на следующий набор коэффициентов *(1, 1^2, ..., 1^t)*. Однако при замене всех константных значений поля Галуа, используемых в модуле поиска Ченя, на обратные, получается, что первая ветвь будет умножать коэффициенты полинома локатора ошибок на набор коэффициентов *(α^{-1}, α^{-2}, ..., α^{-t}),* то есть вместо значения *$\Lambda(1) = \Lambda(\alpha^{2\text{^}m-1})$* будет получаться значение *$\Lambda(\alpha^{-1}) = \Lambda(\alpha^{2\text{^}m-2})$*.

На первом цикле обратного поиска Ченя на выходе ожидается набор значений (*$\Lambda(1)$, $\Lambda(\alpha^{-1})$, ..., $\Lambda(\alpha^{-p-1})$)*. При простой замене на обратные всех констант поля Галуа, используемых в модуле поиска Ченя, на выходе модуля на первом цикле получается набор значений (*$\Lambda(\alpha^{-1})$, $\Lambda(\alpha^{-2})$, ..., $\Lambda(\alpha^{-p})$)*. Решить эту проблему можно путем умножения коэффициентов полинома локатора ошибок на набор коэффициентов *(α^1, α^2, ..., α^t)* в начале расчета. Назовем данный набор коэффициентов *initConstValSet*.

В случае реализации универсального БЧХ декодера, использующего различные поля Галуа, количество наборов коэффициентов *initConstValSet* будет равно количеству используемых полей Галуа. Так, для БЧХ декодера стандарта DVB-S2 количество используемых полей Галуа равно 2, а количество режимов работы равно 21. Для БЧХ декодера стандарта DVB-S2X количество полей Галуа равно 3, а количество режимов работы равно 49. Таким образом, для учета укороченности БЧХ кода при использовании прямого поиска Ченя необходимо хранить 21 (DVB-S2) или 49 (DVB-S2X) наборов коэффициентов *(α^u, α^{2u}, ..., α^{tu})*, а для обратного поиска Ченя – 2 (DVB-S2) или 3 (DVB-S2X) набора коэффициентов *(α^1, α^2, ..., α^t)*.

Результаты синтеза

Для оценки эффективности обратного поиска Ченя необходимо провести синтез модуля. Для проведения синтеза была выбрана ПЛИС фирмы Xilinx Kintex-7. Были синтезированы следующие вариации модуля поиска Ченя: прямой поиск Ченя без учета укороченности БЧХ кодов (рисунок 2), прямой поиск Ченя с учетом укороченности БЧХ кодов (рисунок 3) и обратный поиск Ченя (рисунок 4). Модуль поиска Ченя был реализован для БЧХ декодера стандарта DVB-S2X с количеством режимов работы 49, а также с коэффициентом распараллеливания поиска Ченя равного 8. Результаты синтеза приведены в таблице 1.

Таблица 1 – Результаты синтеза модуля поиска Ченя

Вариант реализации	Учет укороченности БЧХ кодов	Тип поиска Ченя	Затраты ресурсов ПЛИС	
			LUT	FF
1	Не учитывается	Прямой	5389	2319
2	Учитывается	Прямой	6634	2522
3	Учитывается	Обратный	6449	2420

Как видно из таблицы 1, применение обратного поиска Ченя с набором константных значений *initConstValSet* позволяет уменьшить затраты ресурсов ПЛИС, примерно, на 200 LUT и 100 FF. Дальнейшим направлением исследования является попытка избавиться от константного набора *initConstValSet* для обратного поиска Ченя.

К минусам обратного поиска Ченя можно отнести то, что на выход БЧХ декодера будет поступать кодовое слово в обратном порядке. При реализации обратного поиска Ченя, необходимо учитывать, чтобы последующий блок обработки был способен работать с перевернутым кодовым словом.

Библиографический список

1. Zhou Chen, Yulong Zhang, Yan Ying, Chuan Wu, & Xiaoyang Zeng. (2009). An area-efficient and degree-computationless BCH decoder for DVB-S2. 2009 IEEE 8th International Conference on ASIC. doi:10.1109/asicon.2009.5351625

2. Yanni Chen, & Parhi, K. K. (n.d.). Area efficient parallel decoder architecture for long BCH codes. 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing. doi:10.1109/icassp.2004.1327050

3. Wu, T. (2014). Efficient Recording of Parallel Chien Search Results of BCH Code by Three-Staged and Group-Sorted Circuit. 2014 IEEE 17th International Conference on Computational Science and Engineering. doi:10.1109/cse.2014.329

4. С.С. Владимиров, Математические основы теории помехоустойчивого кодирования – СПб ГУТ – стр. 31 – 2014.

УДК 004.89; ГРНТИ 20.51.21

ЦИФРОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ РЕАЛЬНОГО ВРЕМЕНИ

С.С. Ласкунов, У.С. Гарцуева

Белорусский национальный технический университет,
Республика Беларусь, Минск, Laskunov.stas@mail.ru ylgarc@gmail.com

Аннотация. В работе рассматриваются виды цифровых технологий. Приводятся их основные особенности, а также принципы их работы.

Ключевые слова: цифровые технологии, информационные технологии, искуственный интеллект, информация, цифровая система.

REAL TIME DIGITAL INFORMATION TECHNOLOGIES

S.S. Laskunov, U.S. Gartsueva

Belarusian National Technical University,
Belarus, Minsk, Laskunov.stas@mail.ru ylgarc@gmail.com

The summary. The paper considers the types of digital technologies. Their main features are given, as well as the principles of their work.

Keywords: digital technologies, information technologies, artificial intelligence, information, digital system.

Информационные технологии — это общий термин для различных технологий, используемых в основном для управления и обработки информации. Он в основном применяет информатику и коммуникационные технологии для проектирования, разработки, установки и внедрения информационных систем и прикладного программного обеспечения .Его также часто называют информационными и коммуникационными технологиями (ИКТ). В основном это сенсорные технологии, компьютерные и интеллектуальные технологии, коммуникационные технологии и технологии управления. В более узком смысле включает в себя все технические ресурсы, которые служат для создания, хранения, архивирования и использования цифровой информации. IwS также включает передачу информации с использованием коммуникационных технологий.

Применение информационных технологий включает компьютерное оборудование и программное обеспечение, сетевые и коммуникационные технологии, средства разработки прикладного программного обеспечения и т. д. С момента популярности компьютеров и Интернета люди все чаще используют компьютеры для производства, обработки, обмена и

распространения различных форм информации (таких как книги, деловые документы, газеты, записи, фильмы, телепрограммы, голос, графика, изображения и т. д.). .).

В компаниях, школах и других организациях архитектура информационных технологий представляет собой всеобъемлющую структуру для внедрения и развития информационных технологий для достижения стратегических целей. Он включает в себя управленческую и техническую составляющие. Его компонент управления включает миссию, функции и требования к информации, конфигурацию системы и поток информаций, технический компонент включает стандарты и правила информационных технологий для реализации структуры системы управления. Поскольку компьютеры занимают центральное место в управлении информацией , компьютерные отделы часто называют «отделами информационных технологий». Некоторые компании называют это подразделение «Информационными службами» (IS) или «Информационными службами управления» (MIS). Другие компании предпочитают отдавать отдел информационных технологий на аутсорсинг, чтобы получить больше преимуществ.

Виды цифровых технологий и их свойства

1. Искусственный интеллект и машинное обучение, глубокое обучение.

На основе данных об особенностях интеллекта и расшифровках аспектов обучения машина может воспроизвести эти процессы.

Выделяют три группы систем искусственного интеллекта:

А) ограниченный искусственный интеллект — решение одной конкретной задачи;

Б) общий искусственный интеллект — может выполнять много задач как человеческий мозг;

В) сверхразумный искусственный интеллект — выше интеллекта человека.

Машинное обучение — направление искусственного интеллекта, включающее методы, с помощью которых можно обучить. Машины получают данные и обучаются по ним. Например, решение класса задач на распознавание образов.

Глубокое обучение — подмножество машинного обучения – использует нейронные сети для решения реальных задач. Нейронные сети имитируют человеческое поведение в процессе принятия решений.

2. Компьютерное зрение — область искусственного интеллекта, направленная на анализ видео и изображений. Компьютер наделяют набором методов, с помощью которых он извлекает информацию из увиденного. Машины могут обнаруживать, отслеживать и классифицировать объекты.

3. Нейросети — математическая модель, программа которой функционирует подобно мозгу живого организма. Сущность нейросетей заключается в построении программы по принципу функционирования биологических нейронных сетей. С помощью нейронных сетей решают задачи классификации, предсказания, распознавания.

4. Блокчейн и криптовалюты.

Информацию преобразовывают в битовую строку фиксированной длины — хешируют данные. В каждом блоке блокчейна хранится информация о предыдущем блоке. Например, база данных, в которой содержится список сотрудников с транзакциями в системе, краткими характеристиками информационных процессов.

Криптовалюта — разновидность цифровой валюты. Ее количество определяется количеством данных расчетных единиц, которое записывается в соответствующей позиции информационного пакета протокола передачи данных.

5. Большие данные или Big Data — структурированные или неструктурированные массивы данных большого объема. Их используют для статистики, анализа, принятия решений.

6. Телемедицина — обмен медицинской информацией посредством использования компьютерных и телекоммуникационных технологий. Также дистанционное предоставление списка медицинских услуг.

7. Виртуальная и дополненная реальность.

Виртуальная реальность — искусственно созданная трехмерная цифровая среда, нацеленная на передачу человеку информации через его органы восприятия. С помощью сенсорных устройств человек может погрузиться в интерактивный мир.

Виртуальная реальность — правдоподобная, интерактивная, изучаемая, с эффектом присутствия. Специалисты тщательно изучают психологию будущих пользователей. Связь с реальным миром отсутствует.

8. Кибербезопасность — комплекс мер безопасности для обеспечения защиты и конфиденциальности, целостности и доступности данных.

Основные категории кибербезопасности:

1. безопасность сетей;
2. безопасность приложений;
3. безопасность информации;
4. операционная безопасность;
5. аварийное восстановление и непрерывность бизнеса;
6. повышение осведомленности.

Аппаратное обеспечение, программное обеспечение и коммуникационные сети необходимы для обработки информации и данных. Все они объединяются под общим термином ИТ. В компаниях эти технические ресурсы являются частью информационных систем. В целом, ИТ и их использование широко распространены в компаниях, а также в государственном и частном секторах. Он играет центральную роль в банках, так как они интенсивно работают с цифровой информацией. Контракты, движения по счетам и тому подобное хранятся или передаются в цифровом виде. ИТ-безопасность и доступность информационных систем особенно важны для банков, поскольку, с одной стороны, конфиденциальные данные клиентов хранятся и обрабатываются, а с другой стороны, сбои в неблагоприятных случаях могут привести к огромным потерям (например, при торговле акциями). В районе Онлайн-банкинг должен быть обеспечен с помощью ИТ, например, в форме генераторов PIN и TAN, чтобы гарантировать подлинность онлайн-переводов. В этом контексте также необходимо обеспечить соответствие информационных систем требованиям безопасности аудита и другим нормативным актам и законам.

Особенности цифровых технологий

Появление цифровых технологий коренным образом изменило понимание людьми сигналов и систем.

Цифровая технология имеет следующие характеристики:

1. Цифровые сигналы формируются путем введения двоичных значений.

2. Цифровые сигналы удобны для компьютерной обработки алгоритмов, легко реализовать управления процессом и отображение информации. Цифровая техника неотделима от компьютеров.

3. За счет использования логики двоичного уровня цифровые устройства обладают определенной степенью отказоустойчивости, могут эффективно подавлять влияние шумов и различных сигналов помех и являются более надежными, чем аналоговые устройства.

4. Точность цифровой системы можно повысить за счет увеличения количества двоичных разрядов и метода. Аналоговые сигналы легко искажаются шумом во время передачи, и точность системы часто невысока, в то время как цифровым системам нужно только увеличить количество квантованных битов и уменьшить квантованный шум для получения удовлетворительной точности.

5. Цифровые технологии улучшают использование ресурсов. Используя превосходную технологию цифровой модуляции, скорость использования полосы частот канала может достигать кодовой скорости в несколько бит/с на полосу пропускания в Гц. Технология сжатия цифровой информации может уменьшить скорость передачи данных, пропускную способность канала и объем носителя без искажения или меньшего искажения.

6. Компоненты цифровой системы обладают высокой универсальностью и хорошей масштабируемостью. Представленный микропроцессором, он может выполнять различные функции только при поддержке другого программного обеспечения. Это сопобствует стандартизации аппаратного обеспечения, реализует многоцелевое назначение одной машины и расширяет функции и эффективность использования системы. После оцифровки различная информация образует битовый поток, который можно передавать и обрабатывать совместно, реализуя таким образом мультимедийную технологию.

Из-за вышеупомянутых характеристик цифровых технологий цифровые продукты сильно отличаются от традиционных аналоговых продуктов. Характеристики цифровых продуктов:

1. миниатюризация;
2. высокая надежность;
3. высокая стоимость;
4. многофункциональность;
5. интеллектуальность;
6. удобство;
7. быстрота работы.

Использование информационных технологий

ИТ используются в коммерческих целях с начала 1960-х годов. Первоначально компании использовали мейнфреймы для централизованной обработки больших объемов данных. С распространением персональных компьютеров в 1980-х годах использование ИТ увеличилось в частном, деловом и государственном секторах. Интернет появился в начале 1990-х годов и сделал возможным создание глобальных сетей и сотрудничество с использованием ИТ. В 2000 году рост ИТ-индустрии достиг своего пика. Поскольку высокие ожидания, возлагаемые на ИТ и ИТ-компании, не оправдались, лопнул так называемый пузырь доткомов. Однако эта неудача не смогла предотвратить рост цифровизации. ИТ позволили внедрить новые бизнес-модели в банковском секторе, включая онлайн-банкинг.

Сегодня Интернет, а также услуги и предложения, предоставляемые в нем, распространены повсеместно, и их также можно использовать с помощью мобильных устройств, таких как смартфоны. Использование ИТ сегодня принципиально не ставится под сомнение, но предпринимаются попытки использовать ИТ для получения конкурентных преимуществ. С одной стороны, ИТ может способствовать внедрению новых бизнес-моделей и процессов; с другой стороны, использование ИТ может помочь компаниям быстрее и эффективнее управлять своими существующими процессами. Например, кредитоспособность клиентов можно проверить с помощью ИТ-методов оценки и основных данных клиентов. В дополнение к различным функциям сети также обеспечивают доступ определенного оборудования и программного обеспечения или целых систем к одним и тем же данным через центральный компьютер, так называемый сервер, с целью обмена данными. В компаниях подключение к

серверу обычно осуществляется через LAN (локальная сеть) или WLAN (беспроводная локальная сеть).

Обработка данных и информации особенно заслуживает защиты. Под ИТ-безопасностью понимаются меры, защищающие системы от несанкционированного доступа третьих лиц. Меры в основном включают резервное копирование данных, шифрование, контроль доступа и настройку ограниченных учетных записей пользователей.

Межсетевые экраны – как специализированные аппаратные или программные компоненты – обеспечивают возможность чтения и оценки трафика данных в компании. В соответствии с индивидуально предопределенными правилами брандмауэра определяется, какие пакеты данных разрешены, а какие заблокированы.

Согласование ИТ имеет особое значение что гарантирует, что использование ИТ в компании соответствует стратегии и целям компании. Бизнес-процессы и поддерживающие ИТ-системы можно координировать с помощью управления архитектурой предприятия (EAM), которое сначала определяет текущее состояние ИТ-поддержки в компании, выявляет пробелы, а затем закрывает их. Это входит, включая выбор подходящих ИТ, в сферу ответственности руководства ИТ. ИТ-менеджменту приходится переключаться между конфликтующими приоритетами стандартизации и консолидации с целью снижения затрат, с одной стороны, и использованием инновационных решений, позволяющих выделиться на фоне конкурентов, с другой.

В банковском секторе ИТ способствовали внедрению бизнес-модели прямых банков. С 2007 года количество клиентов увеличилось за счет использования ИТ и онлайн-банкинга, в то же время количество отделений сократилось. Таким образом, использование ИТ ведет к экономии средств и должно применяться всеми банками в равной степени, чтобы оставаться конкурентоспособными. Актуальными темами являются FinTechs (финансовые технологии) и криптовалюты на основе ИТ, такие как биткойн.

Библиографический список

1. Цифровые технологии [Электронный ресурс] – Режим доступа: https://wiki.fenix.help/informatika/cifrovye-tehnologii Дата доступа: 14.02.2023
2. Виды информационных технологий [Электронный ресурс] – Режим доступа http://phys.bspu.by/static/lib/inf/posob/stu_m/glaves/glava3/gl_3_4.htm Дата доступа 14.02.2023
3. Использование цифровых технологий [Электронный ресурс] – Режим доступа https://ru.wikipedia.org/wiki/%D0%A6%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D1%8B%D0%B5_%D1%82%D0%B5%D1%85%D0%BD%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D0%B8#:~:text=%D0%A6%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D1%8B%D0%B5%20%D1%82%D0%B5%D1%85%D0%BD%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D0%B8%20%D0%B3%D0%BB%D0%B0%D0%B2%D0%BD%D1%8B%D0%BC%20%D0%BE%D0%B1%D1%80%D0%B0%D0%B7%D0%BE%D0%BC%20%D0%B8%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D1%83%D1%8E%D1%82%D1%81%D1%8F,%D0%B8%20%D0%BC%D0%BD%D0%BE%D0%B3%D0%B8%D1%85%20%D0%B4%D1%80%D1%83%D0%B3%D0%B8%D1%85%20%D1%86%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D1%8B%D1%85%20%D1%83%D1%81%D1%82%D1%80%D0%BE%D0%B9%D1%81%D1%82%D0%B2%D0%B0%D1%85. Дата доступа: 14.02.2023

УДК 004.032.22; ГРНТИ 50.01.81

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ ВЕБ-КОМПОНЕНТОВ ДЛЯ ОРГАНИЗАЦИИ ТЕСТИРОВАНИЯ ПРОИЗВОДИТЕЛЬНОСТИ JAVASCRIPT ФРЕЙМВОРКА «LIT»

В.А. Антошкин*, К.С. Клепиков*, В.И. Щербакова***
Рязанский государственный университет имени С.А. Есенина,
*Российская Федерация, Рязань, *v.antoshkin@rsu.edu.ru, **k.klepikov2408@stud.rsu.edu.ru, ***v.shcherbakova2705@stud.rsu.edu.ru*

Аннотация. В работе обсуждается механизм использования технологии Веб-компонентов для проведения нагрузочного тестирования JavaScript-фреймворка «LIT», разработанного компании Google.
Ключевые слова: JavaScript, компоненты, фреймворк, тестирование, производительность.

USE OF WEB COMPONENTS TECHNOLOGY FOR ORGANIZING PERFORMANCE TESTING OF JAVASCRIPT FRAMEWORK «LIT»

V.A. Antoshkin*, K.S. Klepikov, V.I. Shcherbakova*****
Ryazan State University named after S.A. Yesenin,
*Russia, Ryazan, Рязань, *v.antoshkin@rsu.edu.ru, **k.klepikov2408@stud.rsu.edu.ru, ***v.shcherbakova2705@stud.rsu.edu.ru*

The summary. This article describes the mechanism of using the Web Components technology for load testing of the LIT JavaScript framework developed by Google.
Keywords: JavaScript, components, framework, testing, performance.

«Web Components» – это стандарт, который позволяет создавать пользовательские веб-компоненты на языке программирования JavaScript и использовать их многократно для разработки различных веб-приложений.

Он включает в себя три основных технологии:

1. **Пользовательские элементы** («Custom elements») – это набор средств языка JavaScript, который позволяет регистрировать в браузере пользовательские элементы и задавать их поведение, а затем использовать их при разработке пользовательского интерфейса веб-приложения.
2. **Теневое дерево** («Shadow DOM») – это набор средств языка JavaScript, который позволяет присоединять теневое дерево к пользовательскому HTML-элементу и рендерить его отдельно от основного дерева документа, чтобы избежать возникновения коллизий с другими элементами веб-страницы во время стилизации и управления ими.
3. **HTML-шаблоны** (HTML templates) – это специальные теги «template» and «slot», которые позволяют сформировать определенный шаблон разметки без его непосредственного отображения в веб-документе, а затем многократно использовать для создания внутреннего содержания пользовательских элементов.

Недостатком этого стандарта является сложность его применения на практике, так как это требует от программистов глубоких знаний языка JavaScript и всех тонкостей реализации всех этих трех технологий [1].

Однако существует ряд JavaScript-фреймворков, которые позволяют существенно упростить процесс создания и использования пользовательских компонентов. Одним из таких фреймворков является фреймворк «Lit», который был разработан компанией «Google» в тесном сотрудничестве с командой добровольцев веб-сервиса «GitHub».

Для тестирования его производительности была разработана небольшая программа [2], которая отображает в браузере динамически изменяемые входные данные в виде таблицы с помощью компонента, созданного на основе фреймворка «Lit» (рис. 1).

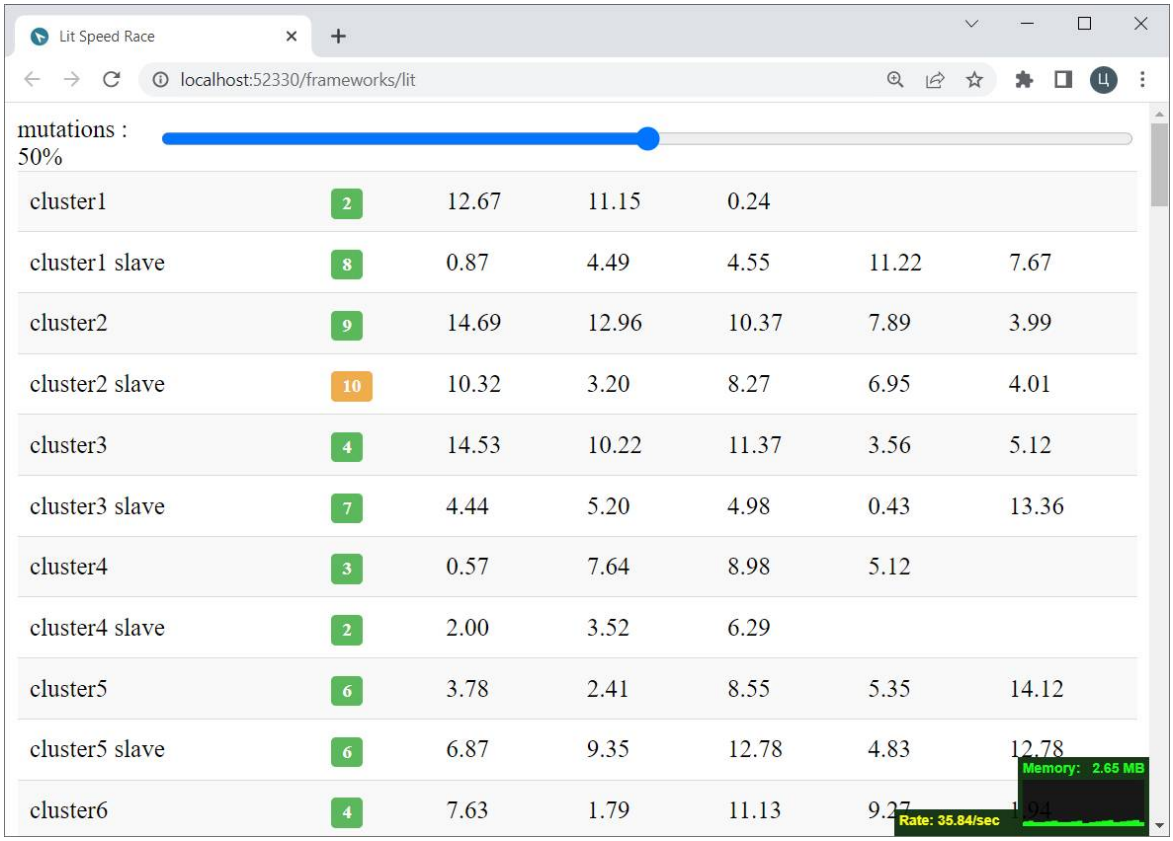


Рис. 1. Интерфейс окна выбора фреймворка

В конструкторе этого компонента (рис. 2) формируется объект базы данных «databases», который и хранит отображаемые данные в виде таблицы.

```
constructor() {
    super();
    this.databases = ENV.generateData(true).toArray();
}
firstUpdated() {
    super.firstUpdated();
    setInterval(() => {
        this.run()
        this.requestUpdate();
    }, ENV.timeout);
}
run() {
    ENV.generateData(true).toArray();
    window.Monitoring?.ping();
}
```

Рис. 2. Основные методы тестируемого компонента

Как только эти данные будут отображены, формируется их новый набор с помощью метода «run». Процесс обновления данных продолжается в постоянном цикле с нулевой задержкой (ENV.timeout = 0), который формируется с помощью функции «setInterval» [3]. На каждом шаге этого бесконечного цикла осуществляет замер времени отображения новых данных с помощью вызова метода «ping». Для реализации этого метода и предложено

разработать специальный веб-компонент, который будет не только вычислять время отображения данных, но и выводить полученные результаты в окне браузера.

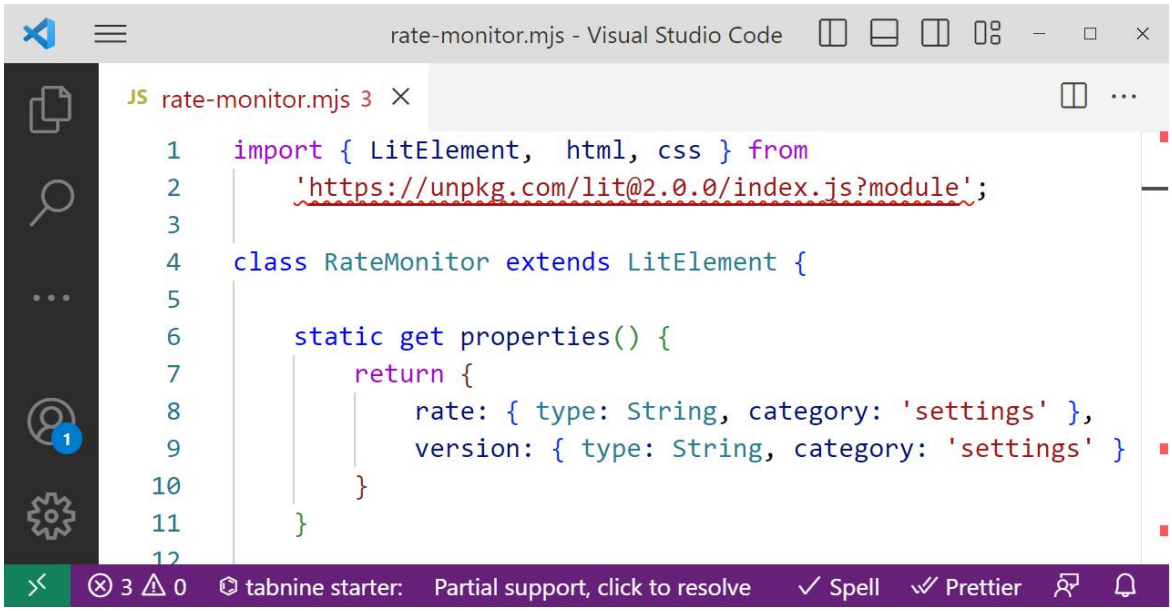


Рис. 3. Список всех тестируемых фреймворков

Чтобы использовать фреймворк для создания веб-компонента, первоначально необходимо подключить его библиотеку с помощью директивы «import» (рис. 3).

Сам компонент реализуется в виде отдельного класса с именем «RateMonitor», который обязательно должен быть наследником базового родительского класса «LitElement», реализованного во фреймворке «Lit» [4]. В самом компоненте задается всего лишь одно реактивное свойство «rate», в котором и храниться последнее сделанное измерение.

Вся логика работы компонента реализуется в его методе «ping» (рис. 4), который и вызывается в момент прорисовки новых данных.

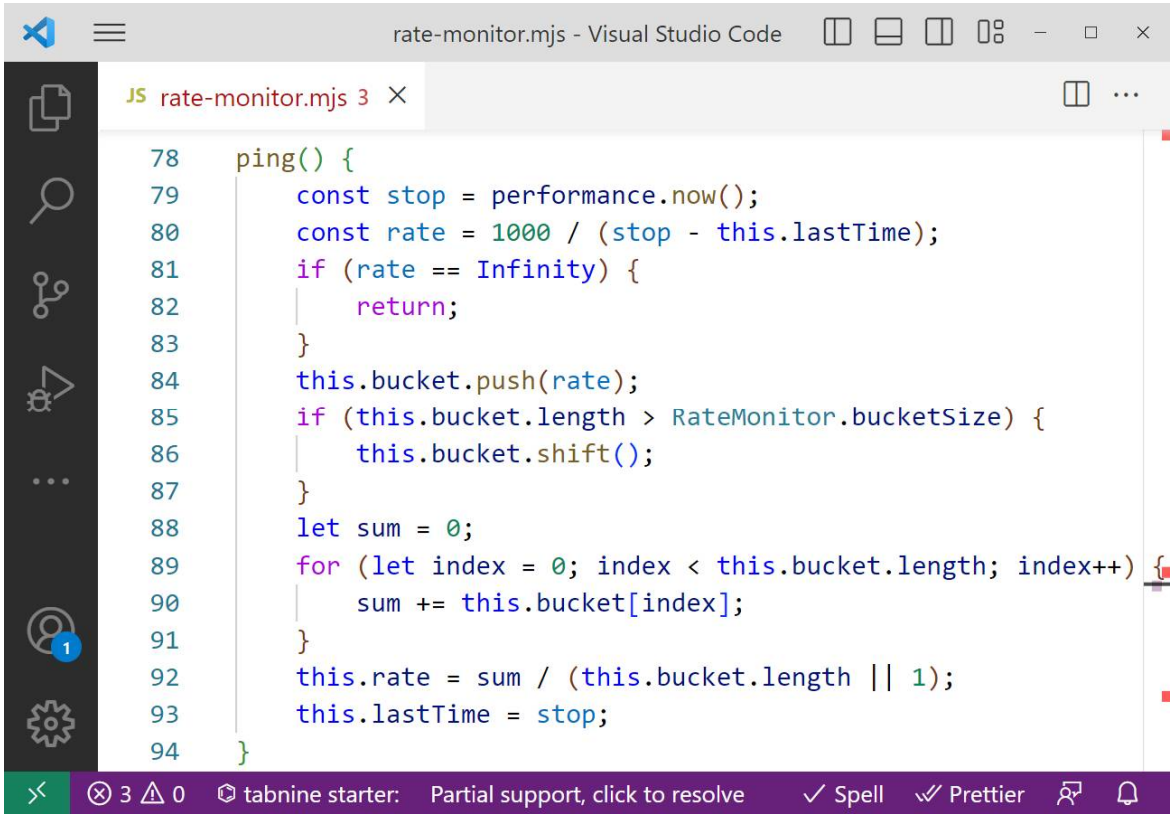


Рис. 4. Список всех тестируемых фреймворков

Здесь с помощью глобального объекта «performance» находится время текущего измерения и по разности со временем предыдущего измерения, находится частота прорисовки фреймворком исходного набора данных. Однако эти измерения носят очень быстро меняющийся характер (рис. 5).

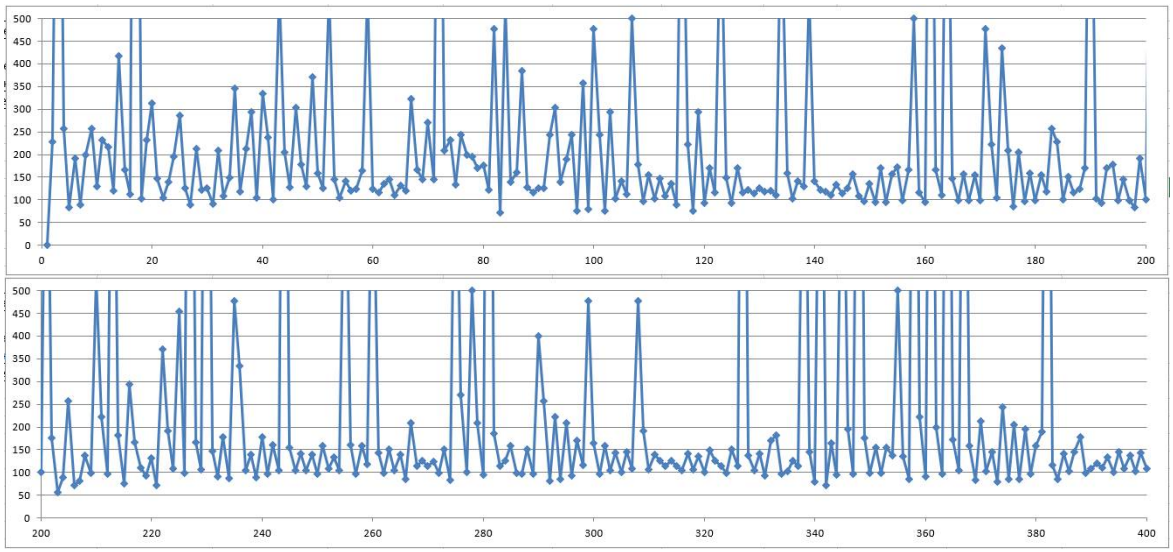


Рис. 5. Характер измерения производительности фреймворка «Lit»

Для того что сгладить полученные результаты, в методе «ping» осуществляется поиск среднего значения по группе измерений. Размер группы был определен опытным путем и выбран на уровне 200, когда среднее квадратическое отклонение становится меньше 50%

(рис. 6).

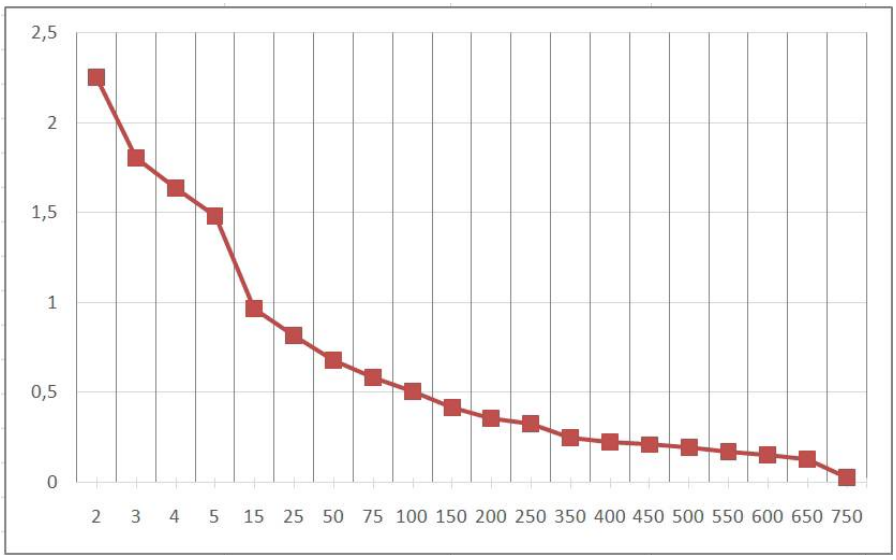


Рис. 6. Зависимость относительного среднеквадратического отклонения от размера группы

В этом случае среднее значение практически остается на одном уровне во всем диапазоне измерения и не зависит от быстро изменяющихся фактических значений (рис. 7).

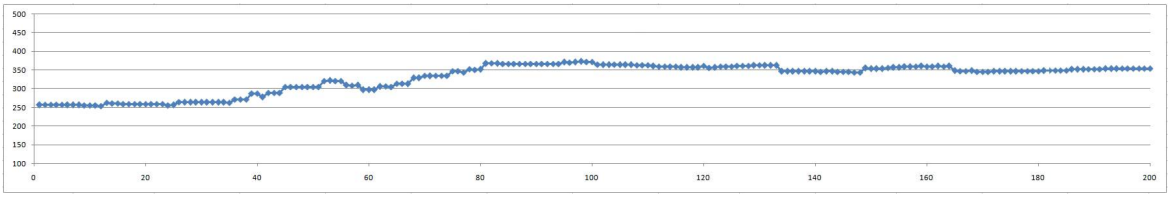


Рис. 7. Характер изменения среднего значения производительности во времени

Вывод найденного среднего значения осуществляется в методе «render» данного компонента (рис. 8).

Рис. 8. Характер изменения среднего значения производительности во времени

Здесь фактически лишь выводится значение свойства «rate» с точности до двух знаков после запятой. Этот метод вызывается фреймворком автоматически при каждом изменении

свойства «rate». В результате этого найденное значение отображается в левом нижнем углу окна браузера во время тестирования производительности фреймворка «Lit».

На рисунке 9 показан итоговый результат тестирования созданного компонента.

Рис. 9. Результат отображения созданного компонента в окне тестирования

На основании сделанного исследования можно сказать, что технологию веб-компонентов можно использовать для создании системы тестирования современных JavaScript-фреймворков, расширяя ее функциональные возможности.

Библиографический список

1. Антошкин В.А., Щербакова В.И. Использование JavaScript интерфейса прикладного программирования для управления NoSQL базой данных «PouchDB» // Современные технологии в науке и образовании – СТНО-2022: Сборник трудов V Международного научно-технического форума. В 10-ти томах, Рязань, 02–04 марта 2022 года / Под общей редакцией О.В. Миловзорова. Том 4. – Рязань: Рязанский государственный радиотехнический университет, 2022. – С. 24-30. – EDN RORRME.

2. Антошкин, В. А., В. И. Щербакова. Тестирование производительности современных javascript фреймворков // Информатика и прикладная математика. – 2022. – № 28. – С. 13-16. – EDN HPNXFT.

3. Антошкин В.А., Иванов А.И. Методика обучения нейросетей с подкреплением на основе игры «Птица в клетке» // Информатика и прикладная математика / – Рязань: Рязанский государственный университет им. С.А. Есенина, 2022. – № 28. – С. 9–12. – EDN MPJYLH.

4. Антошкин, В.А., Иванов А.И. Использование библиотеки Keras для обучения нейросетей выполнению простейших логических операций / В. А. Антошкин, А. И. Иванов // Современные технологии в науке и образовании - СТНО-2022 : Сборник трудов V Международного научно-технического форума. В 10-ти томах, Рязань, 02–04 марта 2022 года – Рязань: Рязанский государственный радиотехнический университет, 2022. – С. 34-40. – EDN OHNBDN.

5. Антошкин В.А., Морозова С.И. Обучение нейронной сети с подкреплением для компьютерной игры «Динозавр T-Rex» // Прикладная математика: современные проблемы математики, информатики и моделирования: Материалы IV Всероссийской научно-практической конференции молодых ученых, Краснодар, 18–23 апреля 2022 года. – Краснодар: ФГБУ "Российское энергетическое агентство" Минэнерго России Краснодарский ЦНТИ- филиал ФГБУ "РЭА" Минэнерго России, 2022. – С. 296-300. – EDN LYVOIH.

СЕКЦИЯ «АКТУАЛЬНЫЕ ЗАДАЧИ ХИМИЧЕСКИХ ТЕХНОЛОГИЙ»

УДК 621.357; ГРНТИ 47.59.49

ИССЛЕДОВАНИЕ КОРИЧНЕВОГО ОКСИДИРОВАНИЯ ДЛЯ ПРОИЗВОДСТВА ПЕЧАТНЫХ ПЛАТ

М.С. Лобанова, Д.А. Маслов
Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, marina040894@gmail.com

Аннотация. В работе рассматриваются коричневое химическое оксидирование, как один из этапов производства печатных плат, его свойства и преимущества перед другими видами оксидирования.
Ключевые слова: оксидирование, печатные платы, оксидные слои.

BROWN OXIDATION RESEARCH FOR PRINTED CIRCUIT BOARD PRODUCTION

M.S. Lobanova, D.A. Maslov
Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, marina040894@gmail.com

Annotation. The paper considers brown chemical oxidation as one of the stages of the production of printed circuit boards, its properties and advantages over other types of oxidation.
Keywords: oxidation, printed circuit boards, oxide layers.

Печатные платы широко применяются во всех сферах нашей жизни, в частности в бытовой технике, мобильных телефонах, компьютерах, медицинском приборостроении, автомобилях, самолетах, космической промышленности, в городском коммунальном хозяйстве, а именно в счетчиках для контроля расхода воды, газа, в приборах для экологического контроля воды, воздуха, земли по различным параметрам и др.

Технологический процесс изготовления печатных плат является сложным многооперационным процессом, представляющий более 50 операций с использованием большого количества оборудования, доходящего до 50 единиц [1].

Одной из проблем в настоящее время является разработка и производство печатных плат, которые соответствуют мировому современному уровню для обеспечения конкурентоспособности, качества и надежности. Один из путей решения данной проблемы является закупка качественных химикатов для установок из других стран, что является очень затратным для производства.

Цель работы – исследовать одну из множества операций производства печатных плат – оксидирование. И рассмотреть возможность подбора доступных импортных реагентов для данного процесса.

Оксидирование при производстве печатных плат необходимо для повышения прочности сцепления внутренних слоев с материалом диэлектрика, а так же для обеспечения хорошей коррозионной устойчивости меди [2]. Пример оксидного покрытия изображен на рисунке 1.

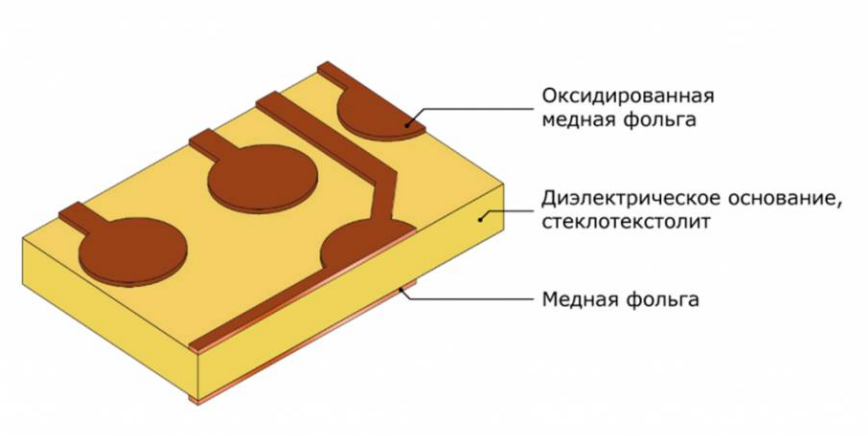


Рис. 1. Оксидное покрытие

На этапе оксидирования верхний слой медной фольги, которая была натравлена ранее, окисляется для лучшей адгезии при последующем прессовании [3].

В зависимости от глубины протекания процесса окисления медь может окисляться до оксида меди (I) красного цвета или до оксида меди (II) черного цвета:

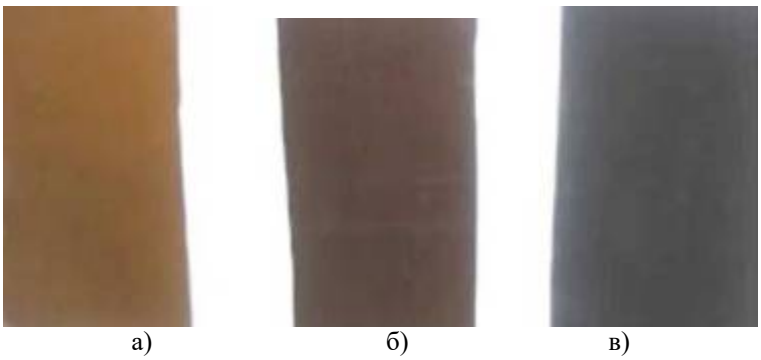
$$8Cu + ClO_2^- + 4\ OH^- \rightarrow 4\ Cu_2O + Cl^- + 2H_2O;$$

$$4Cu + ClO_2^- + 4\ OH^- \rightarrow 4\ CuO + Cl^- + 2H_2O;$$

$$4Cu + S_2O_8^{2-} + 4\ OH^- \rightarrow 2\ Cu_2O + 2SO_4^{2-} + 2H_2O;$$

$$Cu + S_2O_8^{2-} + 2\ OH^- \rightarrow CuO + 2SO_4^{2-} + H_2O.$$

В действительности, оксидированная медная поверхность представляет собой смесь данных окислов. Соотношение красного и черного оксидов в смеси зависит от таких параметров, как температура, время обработки, соотношение окислительного и щелочного компонентов, таким образом, цвет оксидных пленок будет различным. Виды оксидирования показаны на рисунке 2.



а) б) в)

Рис. 2. Различия в цвете оксидных слоев: а) бронзовый оксид, б) коричневый оксид, в) черный оксид

Оксидные слои отличаются друг от друга не только окраской, но и толщиной слоя, кристаллической структурой и адгезией к диэлектрику [2].

Основным дефектом, создающий оксидные слои является «эффект розового кольца» - рисунок 3. Он проявляется уже на готовой плате. Возникает он из-за кислых растворов предварительной обработки, которые воздействуют на оксидные слои. Если они подтравливаются, то вокруг отверстия образуется обнажившаяся медь, то есть «розовое кольцо».

Рис. 3. Дефект «розовое кольцо»

Из трех видов оксидных слоев (бронзовый оксид, коричневый оксид, черный оксид) оптимальным соотношением свойств обладает коричневый оксид, так как у него хорошо развитая поверхность; явно выраженная столбчатая кристаллическая структура; коричневый цвет оксида является гарантией отсутствия дефекта «черный мел»; высокая адгезионная прочность, в отличие от бронзового и черного оксидов; удовлетворительная стойкость к температурным воздействиям и к воздействию кислых растворов [2].

Для увеличения адгезии слоев многослойной печатной платы, повышения стойкости к расслаиванию, а также для уменьшения размерной нестабильности внутренних слоев за счет невысоких температур процесса, рекомендуется использовать трехступенчатый процесс коричневого оксидирования, вместо привычного черного.

Первая ступень - это щелочной раствор, обладающий чистящими свойствами, он подготавливает медную поверхность.

Вторая ступень – активатор, разработан специально для продления жизни третей ступени, чаще имеет идентичный состав, но с меньшей концентрацией

Третья ступень - самая основная дает органо-металлическое покрытие коричневого цвета с высокими адгезионными свойствами [4].

В работе для исследования оксидирования использовалось китайское и немецкое трёхступенчатое оксидирование.

Первым этапом вся химия анализировалась методами титриметрического и фотоколометричекого контроля. Анализы подтвердили, что все компоненты проб в пределах нормы и корректировка химии не требуется.

Вторым этапом выполнялся экспериментальный процесс оксидирования в погружных ванночках из полипропилена. В качестве образцов была выбрана металлическая фольга 3х1см и толщиной 282 мкм, образцы были предварительно взвешены на аналитических весах.

Третий этап – сушка и визуальный контроль. При проведении эксперимента, цвет покрытия у всех трех проб был темно-коричневым, что является нормой. На покрытии не было обнаружено непокрытых участков и «розовых колец».

Четвертый этап – гравиметрический контроль. После сушки и визуального контроля образцы взвешивались на аналитических весах. Величина стравливания рассчитывалась по формуле 1:

$$\Delta W*10000/(s*8.96), \quad (1)$$

где ΔW – потеря веса, разница между весом образца до обработки и после;

s – площадь медной фольги.

Величина стравливания меди должна быть 1,3-1,4 мкм.

В процессе работы посредством эксперимента были выбраны химические реагенты для дальнейшего изучения – китайское оксидирование. Они показали хорошую величину стравливания -1,3 мкм.

Библиографический список

1. Галецкий Ф. Особенности производства печатных плат в России // Электронные компоненты. 2001. № 5. С. 18-26.
2. Сравнительный анализ адгезионных слоев/ Ф. Галецкий, И.Лейтес, Л.Петров,Е. Николаева [Электронный ресурс]- Режим доступа: http://www.rts-engineering.ru/TechPro/tpPP/tpPP_19.shtml
3. Многослойные печатные платы: сквозная металлизация [Электронный ресурс]- Режим доступа: https://www.rezonit.ru/directory/baza-znaniy/tekhnologiya-izgotovleniya-pechatnykh-plat-v-kartinkakh/mnogosloynye-pechatnye-platy-skvoznaya-metallizatsiya/
4. Остроухова О.И Синтез и применение полифторированных ингибиторов коррозии / О.И Остроухова., Прокин Е.В. и др.//Материалы XIV молодежной конференции по органической химии

УДК 543.51; ГРНТИ 31.19.01

РАСЧЕТ ФОРМЫ МАССОВОГО ПИКА ПОСРЕДСТВОМ МЕТОДА РУНГЕ-КУТТЫ ДЛЯ РЕШЕНИЯ ЗАДАЧ АНАЛИТИЧЕСКОЙ ХИМИИ

М.В. Дубков, К.А. Ветшев, А.Д. Рубцова

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, adr1999@list.ru

Аннотация. Представлена к рассмотрению программа для расчета формы массовых пиков посредством метода Рунге-Кутты четвертого порядка. Описана методика определения разрешающей способности пиков, получаемых программным путем.
Ключевые слова: расчет формы массового пика, метод Рунге-Кутты

MASS PEAK SHAPE CALCULATION BY THE RUNGE-KUTTA METHOD FOR SOLVING ANALYTICAL CHEMISTRY PROBLEMS

M.V. Dubkov, K.A. Vetshev, A.D. Rubtsova

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, adr1999@list.ru

The summary. A program for mass peak shape calculating by the Runge-Kutta method of the fourth order is presented for consideration. A technique for determining the resolution of peaks obtained programmatically is described.
Keywords: mass peak shape calculation, Runge-Kutta method

Методы качественного анализа в аналитической химии направлены на определение состава химических веществ и их смесей. Различают химические, физические и физико-химические способы выявления природы различных компонентов. Высокой точностью результатов особенно при идентификации газообразных веществ отличаются последние две группы, среди которых выделяют методы масс-спектрометрии, хроматографии и их комбинации [1].

Масс-спектрометрические методы исследования химического состава веществ обладают рядом преимуществ относительно других способов детектирования. Среди них – возможность идентифицировать целые частицы, а не отдельные ионы. Метод состоит в последовательной ионизации изучаемых молекул, разделении получаемых ионов по значениям отношений массы к заряду и непосредственном детектировании путем измерения электрического тока, образуемого движущимися ионами [2].

Посредством расчета формы массовых пиков можно определять разрешающую способность масс-фильтра, которая, в свою очередь, позволяет идентифицировать любые вещества из таблицы Менделеева при заданных значениях весовых (a, b, c, d) и углового (λ) коэффициентов, согласно уравнению распределения потенциала (1) [3].

$$u(x,y)=\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}A_{mn}x^m y^n \tag{1}$$

где A_{mn} – некоторая постоянная, учитывающая весовые и угловой коэффициенты.

Координаты описывающие траектории движения частицы x и y определяются по формулам (2) и (3) соответственно [3].

$$\frac{\partial u(X,Y)}{\partial x}=\sum_{n=1}^{\infty}\left[\begin{array}{l} a_{2n}\sum_{k=0}^{n}\frac{(-1)^k(2n)!(2n-2k)}{(2n-2k)!(2k)!}X^{2n-2k-1}Y^{2k}+\\ +b_{2n}\sum_{k=0}^{n-1}\frac{(-1)^k(2n-1)!(2n-2k-1)}{(2n-2k-1)!(2k+1)!}X^{2n-2k-2}Y^{2k+1}+\\ +c_{2n}\sum_{k=0}^{n}\frac{(-1)^k(2n+1)!(2k)}{(2n-2k+1)!(2k)!}X^{2k-1}Y^{2n-2k+1}+\\ +d_{2n}\sum_{k=0}^{n}\frac{(-1)^k(2n+1)!(2n-2k+1)}{(2n-2k+1)!(2k)!}X^{2n-2k}Y^{2k}\end{array}\right] \tag{2}$$

$$\frac{\partial u(X,Y)}{\partial y}=\sum_{n=1}^{\infty}\left[\begin{array}{l} a_{2n}\sum_{k=0}^{n}\frac{(-1)^k(2n)!(2k)}{(2n-2k)!(2k)!}X^{2n-2k}Y^{2k-1}+\\ +b_{2n}\sum_{k=0}^{n-1}\frac{(-1)^k(2n-1)!(2k+1)}{(2n-2k-1)!(2k+1)!}X^{2n-2k-1}Y^{2k}+\\ +c_{2n}\sum_{k=0}^{n}\frac{(-1)^k(2n+1)!(2n-2k+1)}{(2n-2k+1)!(2k)!}X^{2k}Y^{2n-2k}+\\ +d_{2n}\sum_{k=0}^{n}\frac{(-1)^k(2n+1)!(2k)}{(2n-2k+1)!(2k)!}X^{2n-2k+1}Y^{2k-1}\end{array}\right] \tag{3}$$

Расчет траекторий движения частиц производится посредством применения метода Рунге-Кутты четвертого порядка. Выбор обусловлен высокой степенью точности расчета.

Методика состоит в следующем: выполняется расчет уточняющих коэффициентов по формулам (4-7).

$$k_1=f(x_n,y_n), \tag{4}$$

$$k_2=f(x_n+\frac{h}{2},y_n+\frac{h}{2}k_1), \tag{5}$$

$$k_3=f(x_n+\frac{h}{2},y_n+\frac{h}{2}k_2), \tag{6}$$

$$k_3=f(x_n+h,y_n+hk_2), \tag{7}$$

где h – шаг расчета.

Далее по итерационной формуле (8) вычисляются приближенные значения в каждой последующей расчетной точке.

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{8}$$

Для расчета формы массового пика необходимо ввести исходные данные: начальный и конечный коэффициент q_u, угловой коэффициент, начальные скорости движения частиц, число периодов (конечное время), шаг для расчета q_u, шаг расчета для метода Рунге-Кутты, а также начальный радиус влета частиц, количество вводимых частиц и весовые коэффициенты. Для каждого q_u создается область заданного диаметра, которая в случайном порядке засеивается указанным числом частиц. Для наглядности в программе отображается последнее засевание области ввода частиц, что представлено на рисунке 1. Начальная фаза также задается случайным образом.

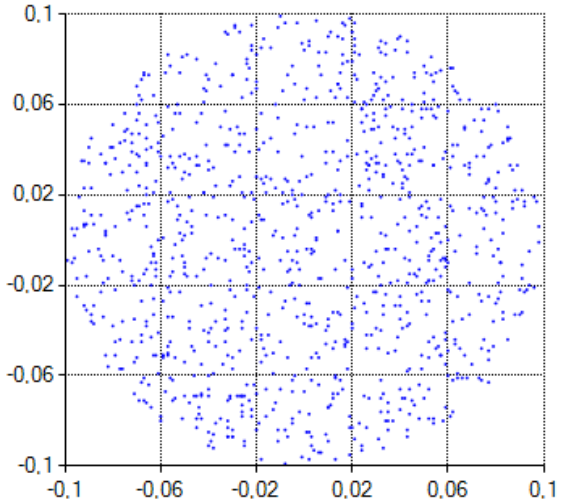


Рис. 1. Изображение случайного засеивания частицами отверстия ввода

По описанной выше методике производится расчет траектории движения частиц при заданном значении q_u для каждой частицы. В случае, если значение рассчитанной координаты X или Y превышает 1, считается, что частица вылетела за пределы спектрометра, расчет прекращается. В случае, если частица пролетает указанное количество периодов, она прибавляется к числу пролетевших частиц при заданном значении q_u.

После расчёта всего диапазона коэффициентов q_u строится график зависимости доли пролетевших ионов от значений этих коэффициентов, что и представляет собой массовый пик. Затем по представленной ниже методике рассчитывается разрешающая способность пика. Для этого определяются две точки на расстоянии ½ от вершины пика, после чего по формуле (9) рассчитывается разрешающая способность [3].

$$R = \frac{\frac{1}{a_2^2}}{\frac{1}{a_{2\,L}^2} - \frac{1}{a_{2\,R}^2}}, \tag{9}$$

где a_2 – точка максимума массового пика;

a_{2L} - точка на середине левой части массового пика;

a_{2R} – точка на середине правой части массового пика.

Пример рассчитанного программным путем массового пика представлен на рисунке 2.

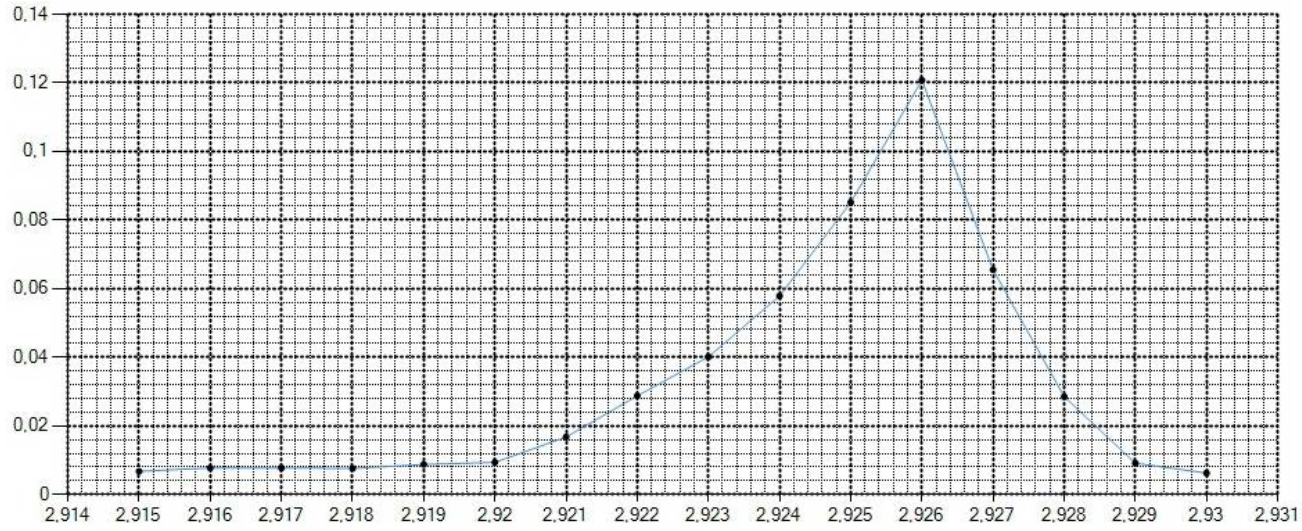


Рис. 2. Форма массового пика, полученного по описанной методике

Зависимость была получена при следующих параметрах:

Начальный коэффициент q_u 2,915
Конечный коэффициент q_u 2,930
Угловой коэффициент λ 1,2444
Начальные скорости движения частиц X’ и Y’ 0
Число периодов (конечное время) 55
Шаг для расчета q_u 0,001
Шаг расчета для метода Рунге-Кутты 0,005
Начальный радиус влета частиц 0,1
Количество вводимых частиц 8000
Коэффициент a_2 1

Подбор значения углового коэффициента λ позволяет получать пики различной формы и разрешающей способности. Значение разрешающей способности равное 200 позволяет идентифицировать практически любые элементы таблицы Менделеева.

В дальнейшем планируется более детальное исследование влияния весовых коэффициентов на форму массового пика и разрешающую способность масс-спектрометра.

Библиографический список

1. Основы аналитической химии. В 2 т. Т. 1 : учеб. для студ. учреждений высш. проф. образования / [Т.А.Большова и др.] ; под ред. Ю.А.Золотова. — 5-е изд., стер. — М. : Издательский центр «Академия», 2012. — 384 с.

2. Основы метода масс-спектрометрии. Практическое применение метода : учебное пособие / Е. А. Илларионова, И. П. Сыроватский ; ФГБОУ ВО ИГМУ Минздрава России, Кафедра фармацевтической и токсикологической химии. – Иркутск : ИГМУ, 2021. – 49 с.

3. Дубков, Михаил Викторович. Синтез масс-селективных электродных структур с возмущениями электрического поля: диссертация ... доктора технических наук : 01.04.01 / Дубков Михаил Викторович; [Место защиты: Рязан. гос. радиотехн. ун-т].- Рязань, 2018.- 299 с.

УДК 621.763; ГРНТИ 53.39.31

АНАЛИЗ СПОСОБОВ УВЕЛИЧЕНИЯ СКОРОСТЕЙ КРИСТАЛЛИЗАЦИИ ГРАНУЛ ВЫСОКОПРОЧНЫХ АЛЮМИНИЕВЫХ СПЛАВОВ

М.Н. Ушакова

Московский авиационный институт (Национальный исследовательский университет)
Российская Федерация, Москва, ushakova.msk@mail.ru

Аннотация. В работе подробно рассматриваются различные схемы и методы получения гранул алюминиевых сплавов, которые позволяют увеличить скорость кристаллизации. Анализируется влияние интенсивности теплоотвода и скоростей кристаллизации на механические свойства материала получаемых полуфабрикатов.
Ключевые слова: гранулы, порошковая металлургия, металлургия гранул, механические свойства материала, скорость кристаллизации, скорость теплоотвода, интенсивность охлаждения, критерии качества гранул.

ANALYSIS OF METHODS FOR INCREASING THE RATES OF CRYSTALLIZATION OF GRANULES OF HIGH-STRENGTH ALUMINUM ALLOYS

M.N. Ushakova

Moscow Aviation Institute (National Research University)
Russian Federation, Moscow, ushakova.msk@mail.ru

The summary. The paper discusses in detail various schemes and methods for obtaining aluminum alloy granules, which allow increasing the crystallization rate. The effect of heat removal intensity and crystallization rates on the mechanical properties of the material of the resulting semi-finished products is analyzed.
Keywords: granules, powder metallurgy, granule metallurgy, mechanical properties of the material, crystallization rate, heat removal rate, cooling rate, quality criteria for granules.

Введение

Одной из наиболее важных проблем, стоящих перед современной металлургией гранул алюминиевых сплавов является разработка технологий получения качественного гранулята при максимальных скоростях кристаллизации капель расплава. Из теории и практики металлургии гранул высокопрочных алюминиевых сплавов известно, что при увеличении скорости кристаллизации капель расплава формируется пересыщенный твердой раствор с уникальной ультрадисперсной структурой, что в итоге приводит к увеличению механических свойств материала гранул [1, 2]. Увеличение механических свойств материала гранул, как исходного сырья, ведет к повышению прочностных и других эксплуатационных характеристик материала конечной детали, а именно компактного готового изделия. Применительно к изготовлению конструктивных деталей авиационной техники принципиально важным является любое увеличение прочностных характеристик материал при неизменной его плотности, так как это однозначно отражается на весе готовой конструкции. Однако при увеличении скоростей кристаллизации гранул высокопрочных алюминиевых сплавов необходимо учитывать также необходимость обеспечения стабильности получаемой мелкодисперсной структуры при длительной работе в условиях высоких и/или низких температур, в условиях циклических нагрузок, а также в условиях достаточно высоких динамических нагрузок.

Анализ перспективных способов увеличения скоростей кристаллизации

Рассматривая способы получения гранулированных материалов можно выделить несколько основных способов: механическое измельчение слитка, метод центробежного распыления оплавляемого электрода, методы газовой атомизации, методы центробежного разбрызгивания через отверстия перфорированного стакана, метод разбрызгивания струи расплава вращающимся диском и т.д. На рисунке 1 представлена классификация наиболее ши-

роко распространенных методов получения гранул. Понятно, что методы механического измельчения слитка на порошинки или гранулы "сводят к нулю" все преимущества быстрой кристаллизации, которые являются основой металлургии гранул. Применение данного метода возможно только в том случае, когда получают деталь спеканием частиц порошка, при этом обрабатываются материалы, имеющие нулевую пластичность при высоких температурах и плохо поддающиеся обработке резанием [3].

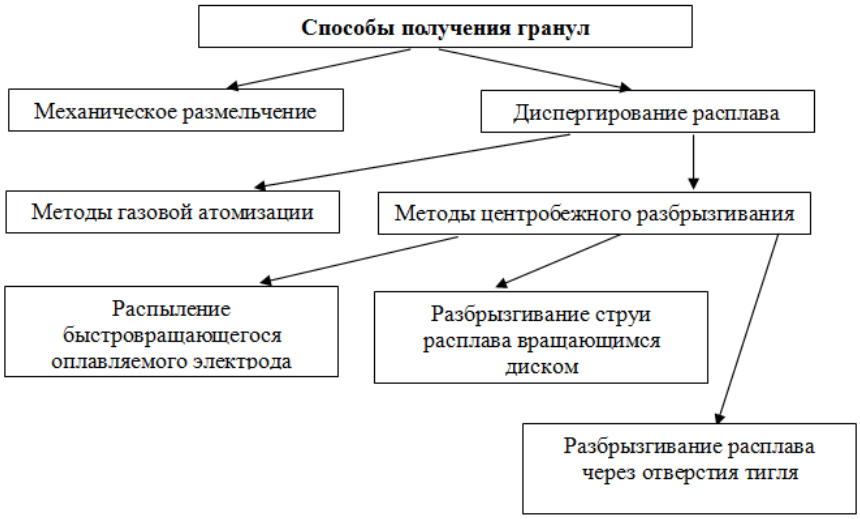


Рис. 1. Классификация наиболее широко распространенных методов получения гранул

Методы газовой атомизации и распыление быстро вращающегося оплавляемого электрода нашли широко применение при получении гранул в европейских странах, США, КНР. Однако данные методы используют охлаждение частиц в инертной среде или на воздухе. Как известно охлаждение в газовой среде, которая характеризуется сравнительно низким коэффициентом теплопроводности, приводит к снижению интенсивности теплоотвода и, следовательно, снижению скорости кристаллизации гранул [4, 5]. Такими методами обычно получают гранулы и порошки дорогостоящих никелевых, титановых сплавов, интерметаллидов алюминида титана, алюминидов никеля типа NiAl и Ni_3Al, гранул сложных сплавов типа Ti-6Al-4V. Вышерассмотренные методы довольно дорогостоящие, так как требуют предварительной подготовки слитка или электрода, используют дорогостоящий инертный газ (наиболее часто в промышленных масштабах производства используют аргон). И что самое главное скорость охлаждения в данных методах довольно низкая. Наиболее действенными методами является либо способ уменьшения размеров получаемой частицы или использование каких либо охлаждающих сред. Однако, уменьшение размеров получаемых частиц до уровня чешуек (50-100 мкм) отрицательно сказывается на последующих операциях спекания или совместной пластической деформации. Мелкие частицы представляют определенные сложности при твердофазном соединении. Кроме того, в случае уменьшения размеров частиц значительно увеличиваются суммарная площадь поверхности порошинок; применительно к алюминиевым сплавам активно формируются окислы Al_2O_3 на поверхности гранул. Пленка Al_2O_3 довольно прочна и эластична, ее необходимо разрушить при последующем спекании и деформации для получения качественного твердофазного соединения. Другое направление интенсивного увеличения скоростей кристаллизации лежит в использовании охлаждающих сред (рис. 2.). И в том и другом случае сохраняется фундаментальная зависимость прочностных характеристик от скорости охлаждения и, следовательно, от скорости кристаллизации [6].

На принципе использования технической воды в качестве охлаждающей среды основаны многие технологии производства гранул высокопрочных алюминиевых сплавов. По сути дела вода играет такую же роль как и в случае обычной закалки. Высокая теплопроводность воды приводит к резкому увеличению теплоотдачи, высокой скорости охлаждения и, следовательно, высокой скорости кристаллизации расплава в виде капли [7, 8].

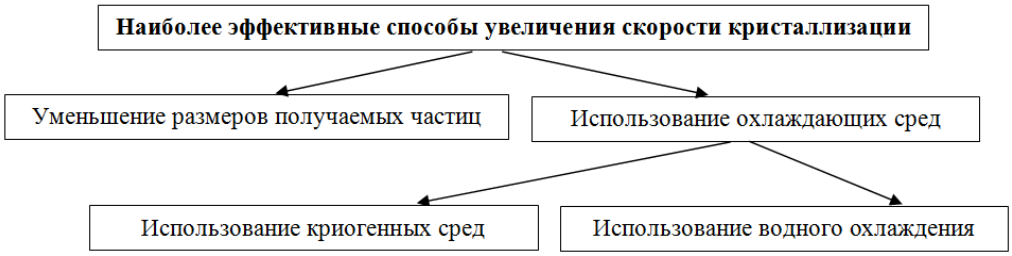


Рис. 2. Наиболее эффективные способы увеличения скоростей кристаллизации гранул

Применение криогенных сред в качестве охладителя еще больше увеличивает скорость кристаллизации гранул, однако такой процесс является довольно дорогостоящим, что делает рассматриваемые производственные процессы крайне не рентабельными.

Таким образом, понятно, что охлаждение капель расплава в воде дает максимальный эффект увеличения скорости кристаллизации. Измерить скорости кристаллизации крайне сложно. Математические модели, по которым можно было бы определить скорость кристаллизации капли металла при центрифуговании расплава в настоящее время пока не разработаны. Однако, можно утверждать, что скорость кристаллизации $v_{об}$ прямо пропорциональна скорости охлаждения $v_{охл}$ и зная скорости охлаждения гранул можно прогнозировать структуру и свойства материала гранул [9].

В частности, из практической деятельности, известны следующие скорости охлаждения при различных диаметрах гранул и особенностях их охлаждения:

- слитки, диаметром 100 мм, охлаждение в воде – скорость охлаждения $v_{охл} = 50\ ^0$C/c;
- гранулы, диаметром 1÷4 мм, охлаждение в воде – скорость охлаждения $v_{охл} = 5\times10^{3}\ ^0$C/c;
- гранулы, диаметром 500 мкм, охлаждение в воде – скорость охлаждения $v_{охл} = 5\times10^{4}\ ^0$C/c;
- гранулы, диаметром 500 мкм, охлаждение на воздухе – скорость $v_{охл} = 5\times10^{2}\ ^0$C/c;
- гранулы, диаметром 50 мкм, охлаждение на воздухе – скорость $v_{охл} = 5\times10^{3}\ ^0$C/c [10].

Видно, что скорость охлаждения при переходе от охлаждения на воздухе к охлаждению в водной среде увеличивает скорость охлаждения $v_{охл}$ и, следовательно, скорость кристаллизации $v_{об}$, в десятки раз. Таким образом, при охлаждении водой гранул диаметром 1-3 мм, наиболее часто применяемых в современной металлургии гранул можно достичь скоростей кристаллизации, характерных для порошинок (чешуек) диаметром 50 мкм [1, 2, 10].

Известно устройство, в котором тонкая струя расплавленного алюминиевого сплава льется на быстровращающийся металлический диск. Диск раздробляет струю и отбрасывает капли расплава в бак, наполненный технической водой; дальнейшее охлаждение происходит в водной среде [11]. Однако в данном методе есть существенный недостаток: кристаллизация расплава капли начинается в момент соприкосновения струи с вращающимся диском и продолжается в процессе полета от диска к воде закалочного бака. Для того, чтобы максимально ускорить процесс теплоотдачи, необходимо максимально сократить время нахождения капли на воздухе. Для этого необходимо минимизировать длину траектории полета капли от основной массы расплава до поверхности охлаждающей среды [12, 13]. Была разработана конструкция устройства [14] и метод получения гранул высокопрочных алюминиевых сплавов [15], который предусматривает наличие вращающегося тигля и вращающегося защитного

стакана с перфорированными отверстиями. Назначение стакана заключается в отбрасывании воды от вращающихся частей устройства путем создания водной воронки при вращении стакана со скоростью 3 000-4 000 оборотов в минуту. Через перфорированные отверстия в тигле за счет центробежных сил происходит деление расплава на капли заданного диаметра, равного диаметру отверстий в тигле. Данное устройство позволяет минимизировать путь, пролетаемый каплей расплава при отделении от основного расплава до контакта с охлаждающей жидкостью, что позволяет утверждать, что вся кристаллизация гранулы происходит в водной охлаждающей среде [14].

Рассматриваемый способ получения гранул заключается в том, что в начальный момент времени капле расплава придается скорость, гарантированно обеспечивающая сбив "паровой оболочки", возникающий вокруг капли расплава от испаряющейся воды. Это позволяет устранить паровую прослойку между каплей расплава и охлаждающей жидкостью и является препятствием для отвода тепла. В итоге, все это обеспечивает увеличение интенсивности теплоотвода и увеличение скорости кристаллизации металлической гранулы [15].

Существуют и другие методы увеличения скорости кристаллизации в водной среде. К ним можно отнести незначительное увеличение скорости поступательного движения капли, создание водных завихрений в охлаждающей жидкости с целью сбива паровой оболочки, использование в качестве охладителя потоков проточной воды, использование добавок, повышающих температуру парообразования воды (например, порошкообразного NaCl) [1, 2, 16]. На рисунке 3 представлена классификация различных способов повышения скоростей кристаллизации (скоростей охлаждения) капли в водной охлаждающей среде.

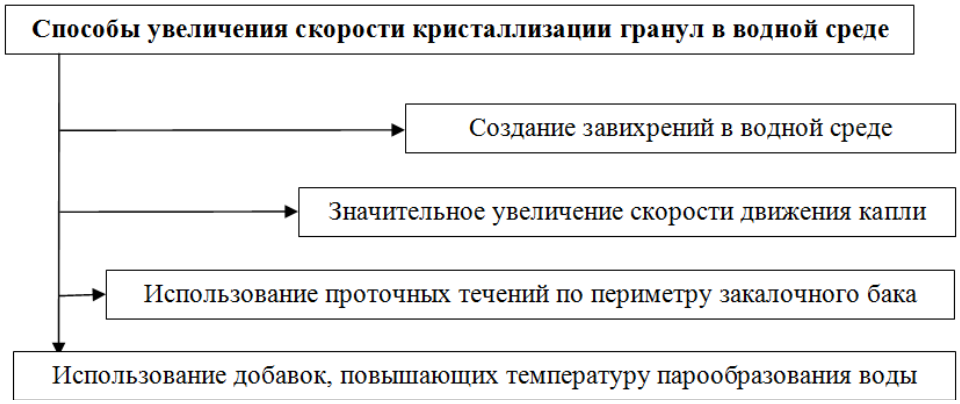


Рис. 3. Способы увеличения скорости кристаллизации гранул в водной среде

Из всех способов интенсификации теплоотвода, представленных на рис. 3 наиболее действенным оказывается способ значительного увеличения первоначальной скорости капли расплава. Он хорошо изучен и показал свою эффективность в процессах получения гранул высокопрочных алюминиевых сплавов системы Al–Cu–Mg (на примере сплавов Д1, Д16) [17] и сплавов системы Al-Zn-Mg-Cu (на примере сплавов В95, В96, В96Ц) [18]. Остальные способы интенсификации теплоотвода в водной охлаждающей среде оказались малоэффективны.

В процессе проведения исследований было установлено, что технология центробежного разбрызгивания расплава через перфорированный тигель с охлаждением в воде довольно хорошо изучена и хорошо работает только при гранулировании алюминиевых сплавов. Возможности применения данного способа гранулирования для других материалов пока мало изучены. Известно, что основными способами производства гранул и порошков более тугоплавких сплавов в отечественной и зарубежной практике являются либо методы газовой атомизации [19], либо методы центробежного разбрызгивания (распыления) вращающегося с большой скоростью оплавляемого электрода [20]. Вышеназванными двумя методами полу-

чают порошки и гранулы сложных сплавов на основе титана и ванадия [21], качественные сферические порошки интерметаллидов алюминида никеля NiAl [22, 23], порошки и гранулы алюминида никеля Ni_3Al [24, 25], а также гранулы жаропрочных никелевых сплавов, использующиеся для изготовления деталей горячего тракта двигателей ГТД [26].

Для этих двух методов представляется возможным только один путь увеличения скоростей кристаллизации, который основан на уменьшении размеров получаемых частиц.

Однако получение гранул высокопрочных алюминиевых сплавов при повышенных скоростях кристаллизации не является самоцелью. Уникальная мелкодисперская структура материала гранул, пересыщенный твердый раствор, повышенное содержание легирующих элементов и высокотемпературных фаз обеспечивает возможность получения путем последующего спекания или высокотемпературной деформационной обработки компактных изделий с более высокими прочностными характеристиками. Так например применение технологий увеличения скоростей кристаллизации гранул сплавов Д1, Д16, В95, В96Ц позволяет получать материалы с повышением прочностных характеристик на 10-15% по сравнению с компактными изделиями из тех же высокопрочных алюминиевых сплавов, получаемых по традиционным технологиям гранулирования [27, 28].

При получении компактных изделий необходимо подбирать такие условия деформирования, чтобы сохранить уникальную структуру и свойства гранул. Кроме того, необходимо разрабатывать системы автоматизированного управления деформационным оборудованием, чтобы в процессе деформации обеспечивалось точное соблюдение температурно-скоростных режимов деформирования [29, 30].

Заключение

В результате проведенного анализа были определены наиболее перспективные способы повышения скоростей кристаллизации высокопрочных алюминиевых сплавов. Определено, что наиболее эффективными способами увеличения скоростей кристаллизации гранул является уменьшение диаметра гранулы или значительное увеличение первоначальной скорости движения гранулы, обеспечивающее эффект сбива "паровой оболочки", неизбежно возникающей вокруг металлической гранулы при ее кристаллизации и охлаждении в водной среде.

Определено, что снижение размеров гранул высокопрочных алюминиевых сплавов до размеров порошинок или чешуек (~ 50 мкм и менее) приводит к определенным технологическим проблемам при их последующей консолидации с получением компактного полуфабриката или готового изделия.

Библиографический список

1. Колпашников А.И., Ефремов А.В. Гранулированные материалы. М.: Металлургия, 1977. – 240 с.

2. Добаткин В.И., Елагин В.И. Гранулируемые алюминиевые сплавы. М.: Металлургия, 1981. – 176 с.

3. Добаткин В. И. Слитки алюминиевых сплавов. Свердловск: Металлургиздат, 1960. – 176 с.

4. Murr L.E., Gaytan S.M. Electron beam melting. Comprehensive materials processing, 2014, Vol. 10, pp. 135-161.

5. Mohanty T., Tripathi B., Mahata T., and Sinha P., Arc plasma assisted rotating electrode process for preparation of metal pebbles, in Discharges and Electrical Insulation in Vacuum (ISDEIV), 2014 International Symposium on, 2014, pp. 741-744.

6. Телешов В.В. Фундаментальная закономерность изменения структуры при кристаллизации алюминиевых сплавов с разной скоростью охлаждения. // Научно-технический журнал "Технология легких сплавов". 2015. № 2, с.13-18.

7. Galkin E.V., Zharov M.V. The prospective technology of production of metal materials grains with extra high rate of solidification. IOP Conference Series: Materials Science and Engineering. 17th International School-Conference "New Materials: Advanced Technologies" IOP Publishing. 1005 (2020) 012020. doi: 10.1088/1757-899X/1005/1/012020.

8. Xia Y., Khezzar L., Alshehhi M., Hardalupas Y.. Droplet size and velocity characteristics of water-air impinging jet atomizer // International Journal of Multiphase Flow. 2017. Vol. 94. P. 31-43.

9. Жаров М.В. Исследование перспективных способов увеличения скорости кристаллизации гранул алюминиевых сплавов // Технология легких сплавов. 2020. № 3. с. 46-53.

10. Жаров М.В. Разработка технологии производства гранулированных материалов с ультрадисперсной структурой из высокопрочных алюминиевых сплавов // Вестник машиностроения, 2022, № 8, с. 49-55. doi: 10.36652/0042-4633-2022-8-49-55.

11. Колпашников А.И., Ефремов А.В., Силин М.Б. Устройство для центробежной грануляции расплава. Авторское свидетельство изобретения. № 403445. Опубликовано 26.10.1973. Бюллетень № 3 – 2 с.

12. Жаров М.В. Исследование свойств гранулированных материалов системы Al-Cu-Mg, прессуемых из гранул, полученных с применением технологии центрифугования при сверхвысоких скоростях охлаждения // Сварочное производство, 2021, № 10 (№ 226), с. 44 – 48. doi: 10.34641/TM.2021.226.4.011.

13. Angelo P. C., Subramanian R. Powder Metallurgy: Science, technology and applications. – PHI Learning Pvt. Ltd., 2008. p. 312.

14. Силин М.Б., Жаров М.В. Устройство для получения металлических гранул. Патент на изобретение RU 2125923 C1, 10.02.1999. Заявка № 97116802/02 от 24.09.1997.

15. Силин М.Б., Жаров М.В. Способ получения металлических гранул. Патент на изобретение RU 2117556 C1, 20.08.1998. Заявка № 97116862/02 от 24.09.1997.

16. Скуратов А.П., Пьяных А.А. Расчетное исследование скорости охлаждения капли алюминиевого расплава в водной среде. // Научные проблемы транспорта Сибири и Дальнего Востока. 2009. № 1. с. 233-235.

17. Жаров М.В. Исследование влияния скоростей кристаллизации гранул на прочностные характеристики прессованных полуфабрикатов алюминиевых сплавов системы Al–Cu–Mg. // Вестник Пермского национального исследовательского политехнического университета. Машиностроение. Материаловедение". 2020. Том 22, № 3. – с. 20-28. doi: 10.15593/2224-9877/2020.3.03.

18. Жаров М.В. Процессы получения гранулированных материалов из алюминиевых сплавов системы Al-Zn-Mg-Cu по технологии сверхбыстрой кристаллизации гранул // Металлург. 2022. № 3. с. 39-49. doi: 10.52351/00260827_2022_03_39.

19. Ario Sunar Baskoro, Sugeng Supriadi, Dharmanto. Review on plasma atomizer technology for metal powder. MATEC Web of Conferences 269, 05004 (2019) IIW 2018. Режим доступа https://doi.org/10.1051/matecconf/201926905004.

20. Жаров М.В. Анализ технологических процессов производства сферических порошков и гранул моноалюминида никеля NiAl для нужд отечественного двигателестроения // Вопросы материаловедения. 2022. № 3 (111). с. 29-40. doi: 10.22349/1994-6716-2022-111-3-29-40.

21. Karlsson J., Snis A., Engqvist H., Lausmaa J. Characterization and comparison of materials produced by electron beam melting (EBM) of two different Ti–6Al–4V powder fractions. Journal of Materials Processing Technology, 2013, Vol. 213 (12), pp. 2109–2118.

22. Жаров М.В. Сравнительный анализ особенностей технологий получения качественного сферического порошка алюминида никеля NiAl // Металлург. 2022. № 11. с. 57–65. doi: 10.52351/00260827_2022_11_57.

23. Галкин Е.В., Жаров М.В. Анализ технических проблем, препятствующих широкому применению сферических порошков из сплавов на основе алюминида никеля NiAl в современной промышленности. В книге: Новые материалы: перспективные технологии получения и обработки материалов. Сборник тезисов докладов 19-й Международной школы-конференции для молодых ученых и специалистов. Москва, 2021. с. 63-64.

24. Каблов Е.Н., Бунтушкин В.П., Базылева О.А. Конструкционные жаропрочные материалы на основе соединения Ni_3Al для деталей горячего тракта ГТД // Технология легких сплавов. 2007. № 2. с. 75-80.

25. Бунтушкин В.П., Базылева О.А., Буркина В.И. Высокотемпературные жаропрочные сплавы на основе интерметаллида Ni_3Al для деталей горячего тракта ГТД // Авиационная промышленность. 2007. с. 41 – 43.

26. Галкин Е.В., Жаров М.В. Формирование заданного структурно-фазового состояния материала гранул жаропрочных никелевых сплавов, получаемых методами газовой атомизации и центробежного распыления литого полуфабриката. В книге: Новые материалы: перспективные технологии получения и обработки материалов. Сборник тезисов докладов 19-й Международной школы-конференции для молодых ученых и специалистов. Москва, 2021. с. 52-53.

27. Жаров М.В. Исследование свойств гранулированных материалов системы Al-Cu-Mg, прессуемых из гранул, полученных с применением технологии центрифугования при сверхвысоких скоростях охлаждения // Технология машиностроения, 2021, № 4 (№ 226), с. 5-9. doi: 10.34641/TM.2021.226.4.011.

28. Жаров М.В. Исследование особенностей кристаллизации гранул высокопрочных алюминиевых сплавов системы Al-Zn-Mg-Cu при сверхвысоких скоростях охлаждения // Вестник Пермского национального исследовательского политехнического университета. Механика. 2021. № 4. с. 71-82. doi: 10.15593/perm.mech/2021.4.08.

29. Жаров М.В. Измерительно-управляющая система термокомпрессионного оборудования с регламентированными температурно-скоростными параметрами деформирования // Измерительная техника. 2022. № 12. с. 46–51. https://doi.org/10.32446/0368-1025it.2022-12-46-51

30. Жаров М.В. Система автоматизированного управления работой термоупругих прессов: решение проблемы инерционности системы // Автоматизация в промышленности. 2021. № 4. с. 31-36. doi: 10.25728/avtprom.2021.04.07.

УДК 665.73; ГРНТИ 61.51.29

ПРИМЕНЕНИЕ ГИБРИДНЫХ МОДЕЛЕЙ ДЛЯ ПОДДЕРЖКИ ПРОИЗВОДСТВЕННОГО ПЛАНИРОВАНИЯ

С.Н. Райович, А.Р. Семенов

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, sezhana.rayovich@gmail.com

Аннотация. В работе рассматриваются возможность усовершенствования процесса производственного планирования путем интеграции машинного обучения с традиционными процессами.

Ключевые слова: машинное обучение, инженерное моделирование, планирование производства, нефтепереработка, гибридные модели.

APPLICATION OF HYBRID MODELS TO SUPPORT PRODUCTION PLANNING

S.N. Rayovich, A.R. Semenov

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, sezhana.rayovich@gmail.com

The summary. The paper considers the possibilities of improving the production planning process by integrating machine learning with traditional processes.

Keywords: machine learning, engineering modeling, production planning, oil refining, hybrid models.

Одной из главных проблем производственного планирования является необходимость производить и перерабатывать различное по качеству сырье, которые будет покрывать спрос. Поэтому каждое предприятие переработки должно обладать гибкостью производственной системы.

Современный этап развития – внедрение методов математического и инженерного моделирования для описания и оптимизации процессов нефтепереработки с целью их оптимизации.

Для создания моделей используют статистические и строгие модели, объединив которые можно получить гибридные модели, которые сочетают в себе все положительные стороны и позволяет увеличивать точность прогнозируемых переменных, как для известных, так и для новых зависимостей.

Создание строгой модели

Для создания строгой модели собирается набор данных по работе установки начиная от режима работы, заканчивая качеством сырья и продуктов в определенный период времени. Далее на основе этих данных строится инженерная модель в любом из имеющихся ПО для инженерного моделирования. Если модель совпадает с фактическими данными с определенной погрешностью, то можно считать, что она описывает работу установки.

Но, если полученные на строгой модели зависимости соотнести с фактическим режимом работы установки, то можно заметить, что модель недостаточно точно отражает работу производственного объекта.

Из плюсов применения строгих моделей можно выделить возможность прогнозирования еще не имеющихся на предприятии процессов.

Создание статистической модели

На основе полученной строгой модели установки формируется исходный набор данных для разработки структуры «База+Дельта». Составление структуры «База+Дельта» позволяет описать сложный нелинейный процесс через систему линейных уравнений при помощи оценки влияния независимых переменных на зависимые.

Так, например, как можно заметить из графиков зависимостей, полученных при моделировании установки низкотемпературной изомеризации при увеличении конца кипения сырья, то есть его утяжелении, выход целевых изопентановой (рисунок 1) и изогексановой (рисунок 2) фракций снижается.

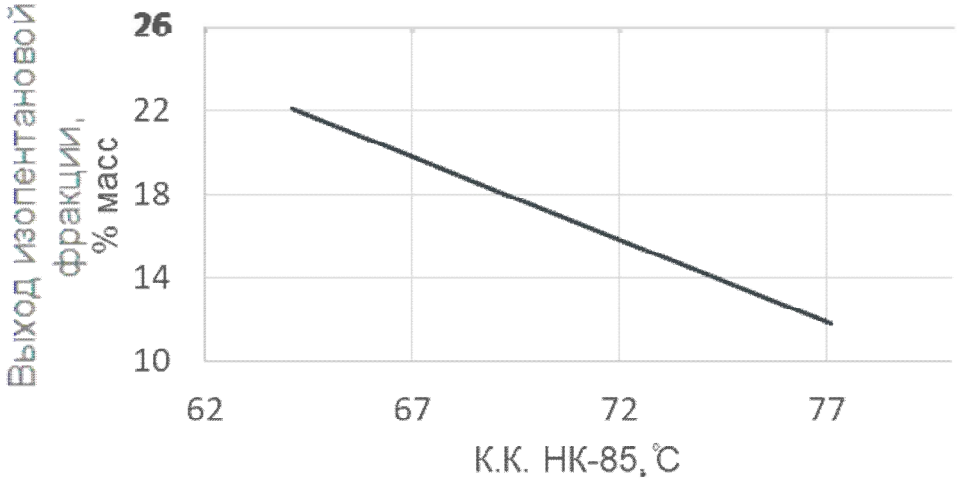


Рис. 1. Зависимость изменения выхода изопентановой фракции от изменения конца кипения НК-85

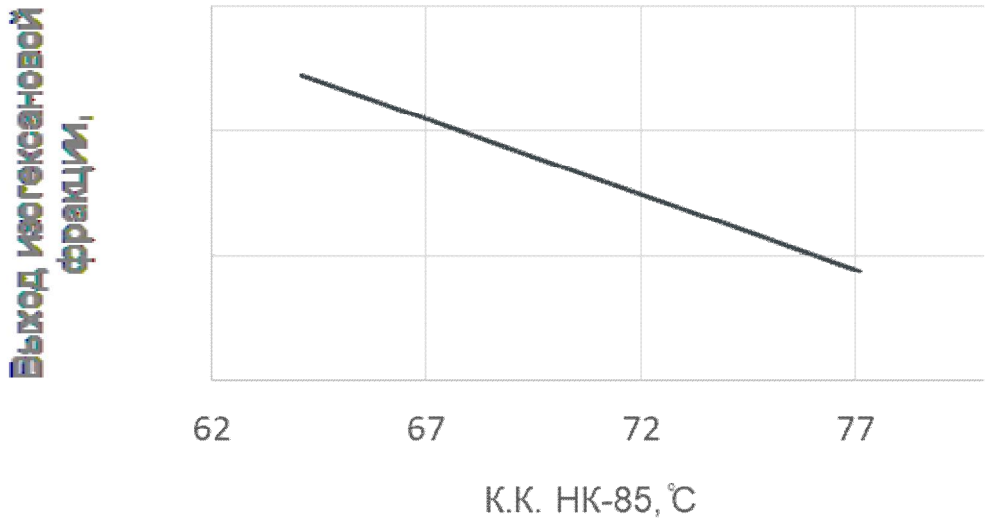


Рис. 2. Зависимость изогексановой выхода изопентановой фракции от изменения конца кипения НК-85

На основе этой структуры строится прогноз работы установки. Пусть имеется некоторое свойство, которое является функцией переменной *k*, а также зависимая переменная продукта X_p, также зависящая от *k*. В таком случае модель «База+Дельта» для расчета выхода записывается в виде уравнения 1.

$$X_p = X_B + \Sigma k_i \cdot (F_i - F_{iB}) \quad (1)$$

где X_p– прогнозируемый параметр;

X_B – базовое значение прогнозируемого параметра;

k_i - зависимость прогнозируемого параметра от изменения *i*-независимой переменной;

F_i– фактическая значение *i*-независимой переменной;

F_{iB}– базовое значение *i*-независимой переменной.

$$k = \frac{\Delta x}{\Delta F} \quad (2)$$

Δx – изменение выхода продукта при изменении независимой переменной;

ΔF – соответствующее изменение независимой переменной.

Оценка качества прогнозирования гибридной модели

Оценка качества прогнозирования проводится на основании среднего абсолютного отклонения фактических показателей работы.

Для оценки сходимости фактических с прогнозируемыми значениями выходов продуктов использовался метод определения среднего абсолютного отклонения прогноза, рассчитываемого по формуле 3.

$$\bar{\Delta} = \frac{1}{n}\sum_{i=1}^{n}\left|X_{p_i} - X_{f_i}\right| \quad (3)$$

X_{p_i} – значение прогноза зависимой переменной;

X_{f_i} – фактическое значение зависимой переменной;

n – Количество значений в выборке.

Таким образом, гибкая структура описания технологических процессов – важнейший инструмент для повышения экономической эффективности планирования и эксплуатации установок. Применение гибридных моделей повышает качество прогнозирования работы технологических объектов, приближая достижение наиболее эффективных режимов работы на установках.

1. Ахметов С.А. Технология глубокой переработки нефти и газа: Учебное пособие для вузов. Уфа: Гилем, 2002. 672 с.

УДК 621.762.2; ГРНТИ 53.39.31

АНАЛИЗ МЕХАНИЗМОВ ОБРАЗОВАНИЯ ДЕФЕКТОВ ФОРМЫ И СТРУКТУРЫ ГРАНУЛ МОНОАЛЮМИНИДА НИКЕЛЯ NIAL ПРИ ИСПОЛЬЗОВАНИИ РАЗЛИЧНЫХ МЕТОДОВ ГРАНУЛИРОВАНИЯ

М.Э. Фролкин

Московский авиационный институт (Национальный исследовательский университет)
Российская Федерация, Москва, frolkin.me@mail.ru

Аннотация. В работе проанализированы механизмы образования различных дефектов формы и структуры гранул никельсодержащих сплавав при их гранулировании. Определены факторы, влияющие на образование дефектов. Определены оптимальные способы и оптимальные режимы для получения качественных гранул алюминида никеля NiAl.
Ключевые слова: металлургия гранул, гранулы, алюминид никеля, дефекты формы, пора, схлопывание поры, поверхностное натяжение, дефекты структуры, сателлиты, качество гранул.

ANALYSIS OF THE MECHANISMS OF FORMATION OF DEFECTS IN THE SHAPE AND STRUCTURE OF NIAL NICKEL MONOALUMINIDE GRANULES USING VARIOUS GRANULATION METHODS

M.E. Frolkin

Moscow Aviation Institute (National Research University)
Russian Federation, Moscow, frolkin.me@mail.ru

The summary. The paper analyzes the mechanisms of formation of various defects in the shape and structure of nickel-containing alloy granules during their granulation. The factors influencing the formation of defects are determined. The optimal methods and optimal modes for obtaining high-quality nickel aluminide NiAl granules are determined.
Keywords: metallurgy of granules, granules, nickel aluminide, shape defects, pore, pore collapse, surface tension, structural defects, satellites, quality of granules.

Введение

Одной из проблем современной металлургии гранул никельсодержащих сплавов является качество получаемых гранул, из которых при последующей обработке горячем изостатическим прессованием или изотермическим спеканием получают готовое изделие. Наличие дефектов формы гранул (таких как поры, усадочные трещины, сателлиты, различные размеры гранул), а также дефекты структуры, вызванные различными скоростями охлаждения (различный химический состав по телу гранулы, впадение фаз и т.д.) по сути дела представляют проблему при последующем изготовлении компактного изделия. Наличие дефекта может быть не устранено при последующем компактировании и спекании и будет являться концентратором напряжений при последующей эксплуатации изготовленной детали. Так как из никельсодержащих сплавов изготавливают только детали ответственного назначения, работающие при повышенных нагрузках и при высоких температурах, то отказ или поломка любой детали подобного класса может привести к потере всего летательного аппарата [1]. Эта проблема касается и гранул алюминида никеля. Алюминид никеля обладает исключительными для жаропрочных материалов механическими и эксплуатационными характеристиками. При его невысокой плотности его прочностные характеристики практически соизмеримых с прочностными характеристикам классических никелевых сплавов. Кроме того, алюминиды никеля характеризуются высокой жаропрочностью и жаростойкостью. Эти уникальные характеристики делают алюминиды никеля крайне перспективными для их использования в качестве материалов или элементов основ композиционных материалов для изготовления деталей современных авиационных двигателей. Однако алюминиды никеля, и, в частности моноалюминид никеля NiAl имеют

существенные недостатки – они практически не обрабатываются резанием. Следовательно, единственной технологией для изготовления деталей из моноалюминида никеля является методы металлургии гранул, когда на последнем этапе из гранул процессами спекания и совместной пластической деформации формируется практические готовое изделие, не требующее какой либо существенной дополнительно механической обработки [2, 3].

Анализ методов получения гранул алюминида никеля

По диаграмме состояния Ni-Al видно, что всего возможно получение пяти видов алюминида никеля ($NiAl_3$, Ni_2Al_3, NiAl (β-фаза), Ni_5Al_3, Ni_3Al (γ′-фаза)). Однако фазы $NiAl_3$, Ni_2Al_3, Ni_5Al_3 имеют низкую температуру разупорядочивания. Поэтому явный практический интерес в качестве высокотемпературных материалов представляют только фазы NiAl и Ni_3Al [4]. В настоящее время наиболее широко изученным является интерметаллид Ni_3Al, который широко применяется при изготовлении различных деталей современных авиационных двигателей, в том числе с применением технологий гранульной металлургии [5, 6]. Однако моноалюминид никеля NiAl является более перспективным материалом. Его плотность на 50% меньше чем плотность интерметаллида Ni_3Al (для сравнения плотность интерметаллида NiAl составляет 5,86 г/см^3, а плотность интерметаллида Ni_3Al – 7,16 г/см^3). Кроме того, согласно диаграмме состояния системы Ni-Al, интерметаллидные соединения моноалюминида никеля NiAl являются самой высокотемпературной фазой в рассматриваемой системе [7].

В настоящее время довольно активно разрабатываются режимы и параметры различных способов получения гранул моноалюминида никеля NiAl. Причем в промышленных масштабах довольно широко используются только два метода: метод плазменного оплавления вращающегося электрода (метод PREP) и метод распыления тонкой струи расплава потоком высокотемпературного инертного газа [8, 9]. Каждый из вышеперечисленных методов имеет свои преимущества и свои недостатки. Отличительной особенностью этих двух методов является высокая производительность процесса, изученность закономерностей процессов при производстве гранул никелевых сплавов и обеспечение возможности производства гранул в среде инертного газа [10]. Однако, несмотря на это имеется ряд серьезных недостатков которые препятствуют широкому распространению металлургии гранул моноалюминида никеля. В первую очередь это проблемы с качеством получаемых гранул. Основная проблема здесь заключается именно в наличии дефектов формы получаемого гранулята [7, 9, 11].

Анализ механизмов образования дефектов при получении гранул моноалюминида никеля

В первую очередь это такие дефекты формы как внутренние поры и наличие сателлитов на поверхности гранулы. Дефекты в виде внутренней поры образуются как в случае распыления оплавляемого электрода, так и в случае диспергирования струей расплава высокотемпературным потоком инертного газа. Суть механизма образования дефекта следующая: при оплавлении вращающегося электрода образуется вытянутая пленка расплава. Чем выше скорость вращения электрода, тем меньше длина формируемой пленки перед ее отделением от основного объема электрода. Под действием центробежных сил пленка отрывается, а далее, в свободном полете за счет поверхностного натяжения схлопывается в сферическую каплю. При этом инертный газ внешней среды оказывается внутри капли расплава, а при последующем охлаждении и кристаллизации, оказывается внутри гранулы [12]. Аналогичный механизм формирования дефектов наблюдается и в

случае применения метода распыления потока расплава моноалюминида никеля инертным газом. При применении данного метода количество гранул имеющих внутреннюю пористость значительно увеличивается. Кроме того, форма подавляющего большинства гранул получаемых методом распыления потоком газа является весьма далекой от сферичности [13]. Даже, несмотря на отсутствие сферичности в гранулах произвольной формы, все равно наблюдаются внутренние поры, заполненные инертной средой. На производстве в подавляющем большинстве случаев в качестве инертной среды используется газ аргон. При дальнейшем формировании готового изделия методом газостатического прессования поры могут самозалечиваться путем выдавливания инертной среды, а могут оставаться в качестве концентратора напряжений. В последнем случае инертный газ не выдавливается и препятствует твердофазному соединению соседних областей гранулы. Пора может оставаться в форме узкой полоски сжатого газа, вытянутой в направлении перпендикулярном направлению главных деформаций при горячем статическом прессовании. Механизмы выдавливания инертного газа или сохранения его внутри гранулы пока еще недостаточно изучены.

Дефекты в виде несферичности гранулы и критически разных размеров гранул также предопределяет определенные проблемы последующей обработке. Эти дефекты в большей степени наблюдаются в случае газовой атомизации, так как скорости охлаждения и кристаллизации в данном случае выше, чем при применении метода PREP и потоки воздуха отрывают разные по величине частицы расплава. Данный процесс никак не контролируется и не управляется. Изменения параметров процесса, а именно температуры перегрева расплава, толщины струи расплава, скорости подачи инертного газа, температуры инертного газа несильно влияют на степень размерного разбега габаритов получаемых гранул [14, 15]. Разнородные по форме и размерам гранулы препятствуют процессу последующей обработки в первую очередь за счет крайне низкой сыпучести гранулята и формированием больших по объему воздушных прослоек при последующей утряске и компактировании. Это влияет на скорость и производительность процесса, а также формирует дефекты в компактном готовом изделии [15,16]. Разные по размерам гранулы приводят и к образованию структурных дефектов. Чем меньше размеры гранулы, тем выше скорость охлаждения и тем выше скорость кристаллизации. Различная скорость кристаллизации приводит к образованию разнородного по химическому составу пересыщенного твердого расплава, формированию различных фаз, выпадению тугоплавких составляющих, появлению нехарактерных интерметаллидов. Причем, эта особенность появления разности структур свойственна не только для алюминидов никеля Ni_3Al и NiAl, но и других жаростойких никелевых сплавов [17, 18]. Доказано, что различная структура гранул при последующей температурной и деформационной обработке формирует различающиеся механические свойства материалов на основе алюминида никеля [19]. Еще одним дефектом формы получаемых гранул является налипание сателлитов на поверхность формирующейся гранулы. Сателлиты представляют из себя меньшие по размеру (иногда в десятки раз) частицы, приварившиеся в процессе кристаллизации к более крупной грануле. За счет разных размеров сателлитов и основных гранул, а также формы получаемой гранулы создаются те же самые проблемы, что и в случае различных размеров и несферичности гранул [20]. Большее количество сателлитов налипших на основные гранулы также характерно при производстве гранул методами газовой атомизации. В случае использования метода PREP количество дефектных гранул (гранул с налипшими сателлитами) значительно ниже [21].

Наиболее перспективным методом получения гранул, в которых отсутствуют все вышерассмотренные дефекты, являются методы центрифугования расплава через

перфорированный стакан. При этом в качестве охлаждающей среды применяется не инертный газ, а водная среда. Так как техническая вода характеризуется большим коэффициентом теплопроводности, чем воздушная среда (более чем в ~10…20 раз), то это позволяет значительно увеличить скорость охлаждения капель расплав и, следовательно, скорость кристаллизации гранул по сравнению с методами PREP и газовой атомизации [22, 23]. В процессе проведения исследований установлено, что с ростом скорости кристаллизации гранул формируется более тонкая дисперсная структура материала, кристаллизуется перенасыщенный твердый раствор и образуется материал с более высокими механическим характеристиками (предел прочности, предел текучести). Эти закономерности были достаточно хорошо изучены для гранулирования методом центрифугования высокопрочных алюминиевых сплавов типа В95, В96, В96Ц [24, 25]. В настоящее время известны методы получения гранул высокопрочных алюминиевых сплавов с повышенными скоростями кристаллизации, когда удается устранить эффект воздушной прослойки, образующейся вокруг кристаллизуемой капли при охлаждении в водной среде. При этом, за счет увеличения интенсивности теплоотвода, удается достичь для гранул диаметром 1-3 мм скоростей кристаллизации равных, скоростям кристаллизации при формировании порошинок размером 10-100 мкм методом газовой атомизации [26, 27, 28]. Однако применение метода центрифугования расплава моноалюминда никеля NiAl довольно затруднено тем, что для получения гранул из этого материала необходимо создать инертную атмосферу. Защитную атмосферу инертных газов необходимо сформировать не только на этапе залива расплава материала во вращающийся перфорированный тигель, но и на этапе выброса капель расплава из отверстий вращающегося перфорированного тигля. Однако самое сложное заключается в том, чтобы сформировать защитную атмосферу при кристаллизации капли в воде, так как возникающей пар при высоких температурах довольно агрессивен по отношению к расплаву моноалюминида никеля [29]. Это затрудняет применение методов центрифугования для получения бездефектных гранул исследуемого материала.

Другая проблема получения качественных изделий из моноалюминида никеля NiAl лежит в области поддержания точных заданных параметров и режимов компактирования и спекания готового изделий из гранул. Как отмечалось выше материал обладает практически нулевой пластичностью, в том числе и при повышенных температурах. Деформирование должно проводиться с точным соблюдением температурно-скоростных параметров, что требует создания специальных автоматизированных систем управления процессами компактирования, спекания и изотермической штамповки гранул моноалюминида никеля [30, 31, 32].

Заключение

Таким образом, в работе был проведен анализ наиболее перспективных методов производства качественных сферических гранул из моноалюминида никеля с точки зрения количества и вида образуемых дефектов. Определено, что наибольшее влияние на качество получаемых готовых компактных изделий оказывает формирование дефектов формы в виде налипания сателлитов на поверхности гранулы, наличие внутренних пор, несферичность формы гранул и различные размеры получаемых гранул. Эти дефекты в результате влияют на плотность засыпки гранул в форму и создают определенные проблемы при преследующем компактировании, спекании или газостатическом прессовании ввиду наличия практически нулевой пластичности моноалюминида никеля NiAl.

Определено, что наиболее перспективным способом получения гранул

рассматриваемого материала является метод центробежного распыления оплавляемого электрода в среде инертного газа. Управление диаметром получаемых гранул и однородностью размеров гранулята возможно путем управления скоростью вращения оплавляемого электрода.

Библиографический список

1. Береснев А.Г., Логунов А.В., Логачева А.И. Проблемы повышения качества жаропрочных сплавов, получаемых методом металлургии гранул // Вестник МАИ. 2008. Т. 15. № 3. с. 83-89.

2. Ночовная Н. А., Базылева О. А., Каблов Д. Е., Панин П. В. Интерметаллидные сплавы на основе титана и никеля. Под ред. Е. Н. Каблова. М.: ВИАМ. 2018. 308 с.

3. Bochenek K., Basista M. Advances in processing of NiAl intermetallic alloys and composites for high temperature aerospace applications // Progress in Aerospace Sciences. 2015. Vol. 79. p. 136-146.

4. Оспенникова О. Г., Базылева О. А., Аргинбаева Э. Г., Шестаков А. В., Туренко Е. Ю. Создание интерметаллидных никелевых сплавов и композиционных материалов на их основе // Вестник МГТУ им. Н.Э. Баумана. Сер. Машиностроение. 2017. № 3. с. 75-89.

5. Бунтушкин В.П., Базылева О.А., Буркина В.И. Высокотемпературные жаропрочные сплавы на основе интерметаллида Ni_3Al для деталей горячего тракта ГТД // Авиационная промышленность. 2007. с. 41 – 43.

6. Каблов Е.Н., Бунтушкин В.П., Базылева О.А. Конструкционные жаропрочные материалы на основе соединения Ni_3Al для деталей горячего тракта ГТД // Технология легких сплавов. 2007. № 2. с. 75-80.

7. Жаров М.В. Сравнительный анализ особенностей технологий получения качественного сферического порошка алюминида никеля NiAl // Металлург. 2022. № 11. с. 57–65. doi: 10.52351/00260827_2022_11_57.

8. Поварова К.Б., Скачков О.А., Казанская Н.К., Дроздов А.А., Морозов А.Е., Макаревич О.Н. Порошковые сплавы NiAl. Получение порошков NiAl // Металлы. 2011. №5. с. 68-78.

9. Xu G. N. Zhang K. F., Huang Z. Q. The synthesis and characterization of ultrafine grain NiAl intermetallic // Advanced Powder Technology. 2012. Vol. 23. p. 366-371.

10. Жаров М.В. Анализ технологических процессов производства сферических порошков и гранул моноалюминида никеля NiAl для нужд отечественного двигателестроения // Вопросы материаловедения. 2022. № 3 (111). с. 29-40. doi: 10.22349/1994-6716-2022-111-3-29-40.

11. Капланский Ю. Ю. Получение узкофракционных сферических порошков жаропрочных сплавов на основе алюминида никеля и их применение в технологии селективного лазерного сплавления. Дис. … канд. техн. наук. М. : МИСиС, 2020. 252 с.

12. Жаров М.В. Разработка технологии производства гранулированных материалов с ультрадисперсной структурой из высокопрочных алюминиевых сплавов // Вестник машиностроения, 2022, № 8, с. 49 – 55. doi: 10.36652/0042-4633-2022-8-49-55.

13. Поварова К.Б., Дроздов А.А., Скачков О.А., Морозов А.Е. Физико-химические подходы к разработке сплавов на основе NiAl для высокотемпературной службы // Металлы. 2011. № 2. с.48-62.

14. Галкин Е.В., Жаров М.В. Анализ технических проблем, препятствующих широкому применению сферических порошков из сплавов на основе алюминида никеля NiAl в современной промышленности. В книге: Новые материалы: перспективные технологии получения и обработки материалов. Сборник тезисов докладов 19-й Международной школы-конференции для молодых ученых и специалистов. Москва, 2021. – с. 63-64.

15. Galkin E.V., Zharov M.V. Analysis of technical problems preventing widespread use of spherical powders from nickel nickel-based alloys NiAl in modern industry. В книге: New materials: advanced technologies for obtaining and processing materials. Сборник тезисов докладов 19-й Международной школы-конференции для молодых ученых и специалистов. Москва, 2021. – с. 65-66.

16. Галиева Э. В. Твердофазное соединение интерметаллидного сплава на основе Ni_3Al и жаропрочного никелевого сплава с использованием сверхпластической деформации. Дис. … канд. техн. наук. Уфа: ФГБУН Институт проблем сверхпластичности металлов, 2021. 195 с.

17. Галкин Е.В., Жаров М.В. Формирование заданного структурно-фазового состояния материала гранул жаропрочных никелевых сплавов, получаемых методами газовой атомизации и центробежного распыления литого полуфабриката. В книге: Новые материалы: перспективные технологии получения и обработки материалов. Сборник тезисов докладов 19-й Международной школы-конференции для молодых ученых и специалистов. Москва, 2021. – с. 52-53.

18. Galkin E.V., Zharov M.V. Formation of given structural-phase state of material of granules of heat-resistant nickel alloys obtaining by methods of gas atomization and centrifugal spraying of cast semi-finished products. В книге:

New materials: advanced technologies for obtaining and processing materials. Сборник тезисов докладов 19-й Международной школы-конференции для молодых ученых и специалистов. Москва, 2021. – с. 54-55.

19. Шевцова Л.И. Структура и механические свойства материалов на основе алюминида никеля, полученных по технологии искрового плазменного спекания порошковых смесей. Дис. … канд. техн. наук. Новосибирск: НГТУ, 2015. 198 с.

20. Жаров М.В. Исследование свойств гранулированных материалов системы Al-Cu-Mg, прессуемых из гранул, полученных с применением технологии центрифугования при сверхвысоких скоростях охлаждения // Сварочное производство, 2021, № 10 (№ 226), с. 44 – 48. doi: 10.34641/TM.2021.226.4.011.

21. Mohanty T., Tripathi B., Mahata T., Sinha P. Arc plasma assisted rotating electrode process for preparation of metal pebbles, in Discharges and Electrical Insulation in Vacuum (ISDEIV), 2014 International Symposium on. 2014. P. 741-744.

22. Galkin E.V., Zharov M.V. The prospective technology of production of metal materials grains with extra high rate of solidification. IOP Conference Series: Materials Science and Engineering. 17th International School-Conference "New Materials: Advanced Technologies" IOP Publishing. 1005 (2020) 012020. doi: 10.1088/1757-899X/1005/1/012020.

23. Ario Sunar Baskoro, Sugeng Supriadi, Dharmanto. Review on plasma atomizer technology for metal powder. MATEC Web of Conferences 269, 05004 (2019) IIW 2018. Режим доступа https://doi.org/10.1051/matecconf/201926905004

24. Жаров М.В. Исследование перспективных способов увеличения скорости кристаллизации гранул алюминиевых сплавов // Технология легких сплавов. 2020. № 3. – С. 46-53.

25. Xia Y., Khezzar L., Alshehhi M., Hardalupas Y.. Droplet size and velocity characteristics of water-air impinging jet atomizer // International Journal of Multiphase Flow. 2017. Vol. 94. P. 31-43.

26. Жаров М.В. Процессы получения гранулированных материалов из алюминиевых сплавов системы Al-Zn-Mg-Cu по технологии сверхбыстрой кристаллизации гранул // Металлург. 2022. № 3. – с. 39–49. doi: 10.52351/00260827_2022_03_39.

27. Жаров М.В. Исследование особенностей кристаллизации гранул высокопрочных алюминиевых сплавов системы Al-Zn-Mg-Cu при сверхвысоких скоростях охлаждения // Вестник Пермского национального исследовательского политехнического университета. Механика. 2021. № 4. – с. 71–82. doi: 10.15593/perm.mech/2021.4.08.

28. Bojarevics V., Roy A., Pericleous K. Numerical model of electrode induction melting for gas atomization // The International Journal for Computation and Mathematics in Electrical and Electronic Engineering. 2011. Vol. 30 (5). pp. 1455-1466.

29. Жаров М.В. Исследование влияния скоростей кристаллизации гранул на прочностные характеристики прессованных полуфабрикатов алюминиевых сплавов системы Al–Cu–Mg. // Вестник Пермского национального исследовательского политехнического университета. Машиностроение, Материаловедение". 2020. Том 22, № 3. – с. 20-28. doi: 10.15593/2224-9877/2020.3.03.

30. Жаров М.В. Измерительно-управляющая система термокомпрессионного оборудования с регламентированными температурно-скоростными параметрами деформирования // Измерительная техника. 2022. № 12. с. 46–51. https://doi.org/10.32446/0368-1025it.2022-12-46-51

31. Алиева Л.И. Образование дефектов в процессах холодного выдавливания // Вестник Херсонского национального технического университета. 2016. № 4 (59). – с. 18-27.

32. Жаров М.В. Система автоматизированного управления работой термоупругих прессов: решение проблемы инерционности системы // Автоматизация в промышленности. 2021. № 4. – с. 31-36. doi: 10.25728/avtprom.2021.04.07.

УДК 678.028.31; ГРНТИ 61.61.13

ВЛИЯНИЕ ТЕПЛОВЫДЕЛЕНИЙ НА РЕЖИМЫ ОТВЕРЖДЕНИЯ ПОЛИМЕРНЫХ КОМПОЗИТОВ

О.С. Дмитриев, А.А. Барсуков, А.О. Дмитриев
Тамбовский государственный технический университет,
Российская академия естественных наук,
Россия, Тамбов, phys@tstu.ru

Аннотация. Рассмотрены негативные явления, возникающие при тепловыделениях и перегреве внутренних слоев плоских толстостенных деталей из полимерных композитов на основе термореактивных связующих. Проведены численные и экспериментальные исследования возникающих экзотермических перегревов, сделано их сравнение. Рассмотрено влияние ограничений наложенных на решение задачи оптимизации режимных параметров процесса отверждения. Проведены расчеты оптимальных циклов отверждения композитов различной толщины.
Ключевые слова: задача оптимизации, полимерные композиты (ПК), режим отверждения, тепловыделения.

INFLUENCE OF HEAT RELEASE ON THE CURING CYCLES OF POLYMER COMPOSITES

O.S. Dmitriev, A.A. Barsukov, A.O. Dmitriev
Tambov State Technical University, Russian Academy of Natural Sciences,
Russia, Tambov, phys@tstu.ru

The summary. The negative phenomena arising from heat release and overheating of the inner layers of flat thick-walled products made of polymer composites based on thermosetting resins are considered. Numerical and experimental studies of emerging exothermic overheatings have been carried out, and their comparison has been made. The influence of restrictions imposed on the solution of the cycle parameters optimization problem of the cure process is considered. Calculations of optimal curing cycles for composites of various thicknesses have been carried out.
Keywords: optimization problem, polymer composites (PC), curing cycle, heat release.

Полимерные композиты (ПК) на основе термореактивных связующих нашли широкое распространение в современной технике. Они заняли особое место среди конструкционных материалов благодаря их уникальным свойствам. Их используют во многих технических отраслях: электро- и радиотехнике, электронике, авиации, судостроении, химической промышленности, автомобильной технике и многих других [1]. Детали из полимерных композитов могут иметь различную толщину и габаритные размеры от мелких и тонкостенных до крупногабаритных. Толщина деталей может быть достигать 50…80 мм.

Технологический процесс изготовления деталей из ПК предусматривает тепловую обработку, в процессе которой активируется термореактивное связующее и происходит химический процесс сшивки полимера или отверждение. Процесс отверждения термореактивных смол и связующих на их основе сопровождается экзотермической реакцией. Выделяемое тепло ускоряет химический процесс и при бесконтрольном проведении процесса отверждения может приводить к негативным последствиям, связанным с температурной деструкцией связующего, аккумулированию внутренних напряжений, короблению готового изделия и т.д. Поэтому процесс отверждения проводят по заранее определенному температурно-временному режиму, при котором негативные проявления экзотермической реакции отверждения сведены к минимуму. Определение такого режима выполняют методами проб и ошибок, по данным дифференциально-сканирующей калориметрии или с помощью математического моделирования и оптимизации [2, 3].

Для решения задачи оптимизации технологического процесса требуется экспериментально с помощью специального оборудования определить параметры математической мо-

дели, исследовать влияние различных факторов на режим отверждения, определить численные значения ограничений, накладываемых на решение задачи оптимизации, и провести расчет режима отверждения. Для исследования процесса отверждения ПК разработана информационно-измерительная система (ИИС), содержащая в своем составе специальные аппаратно-технические средства, а также математическое, алгоритмическое, программное и метрологическое обеспечение [4, 5].

На качество и свойства деталей из ПК оказывают влияние многие факторы: это свойства компонентов композита, межфазные взаимодействия связующего и наполнителя, технологический режим отверждения, способ формования, режим давления, время процесса и т.д. Немаловажную роль в формировании высокого качества деталей из ПК оказывают тепловыделения, вызываемые экзотермической реакцией, так как вследствие низкой трансверсальной теплопроводности ПК отвод тепла от внутренних слоев особенно толстостенной детали затруднен. Это приводит к перегреву внутренних слоев и существенной неоднородности температурного поля $T(x,t)$. Поэтому целесообразно до начала оптимизации режима отверждения провести численные и экспериментальные исследования оценки рисков перегрева вследствие экзотермической реакции процесса отверждения.

Для этого необходимым условием является наличие параметров математической модели процесса отверждения, ИИС и экспериментальная база.

Как пример нами был рассмотрен процесс производства деталей из углепластика. Для расчета и анализа режима отверждения с помощью ИИС нами были исследованы экзотермические температуры. Кроме того, с целью проверки адекватности исследованных характеристик ПК как параметров математической модели отверждения реальному процессу было проведено сравнение значений экзотермической температуры при отверждении $\theta = T_{L/2} - T_0$ в зависимости от толщины L изделия, полученной при численном моделировании и экспериментально измеренных значений. Нами было проведено отверждение 3-х образцов из углепластика в одноступенчатом стандартном регламентном режиме. Образцы для исследования набирали из нескольких слоев препрега, уложенных перекрестной ориентацией волокон. Осуществлялся подъем температуры поверхности со скоростью 3 К/мин до температуры изотермической выдержки. Первый образец толщиной 19,43 мм отверждали при температуре 170°С. Два других образца толщиной 11,07 мм и 17,53 мм отверждали при температуре 190°С. Общее время отверждения составляет t=120 мин. Затем провели серию численных экспериментов моделирования процесса отверждения по таким же режимам для деталей указанных толщин. Результаты расчетов и сравнения θ, полученных экспериментально и расчетным путем приведены на рисунке 1. Разница экзотермических выбросов температур θ, полученных в результате расчета и измеренных экспериментально $\delta = \theta_{расч} - \theta_{эксп}$, лежит в пределах 1,3°С. Максимальные температуры экзотермических выбросов $\max\theta$ при толщине изделия L = 30 мм могут достигать превышения температуры на 24°С, а при толщине изделия 50 мм – на 40°С, что неизбежно приведет к ухудшению прочностных характеристик, вплоть до расслаивания и разрушения изделия.

Таким образом, одноступенчатый стандартный регламентный режим пригоден для производства только тонкостенных пластин до 10 мм толщиной, при котором максимальная температура экзотермического выброса $\max\theta$ не превышает 3°С. В этом случае процесс отверждения протекает при сравнительно однородном температурно-конверсионном поле. Для деталей большей толщины необходимо проведение оптимизационных расчетов режимных параметров отверждения ПК. Эти расчеты особенно необходимы при производстве деталей большой толщины и крупных размеров.

Оптимизация режимных параметров любого процесса начинается с постановки задачи, определяются критерии оптимизации, минимизация которых повышает качество готового изделия, либо повышает производительность, либо уменьшает стоимость изделия.

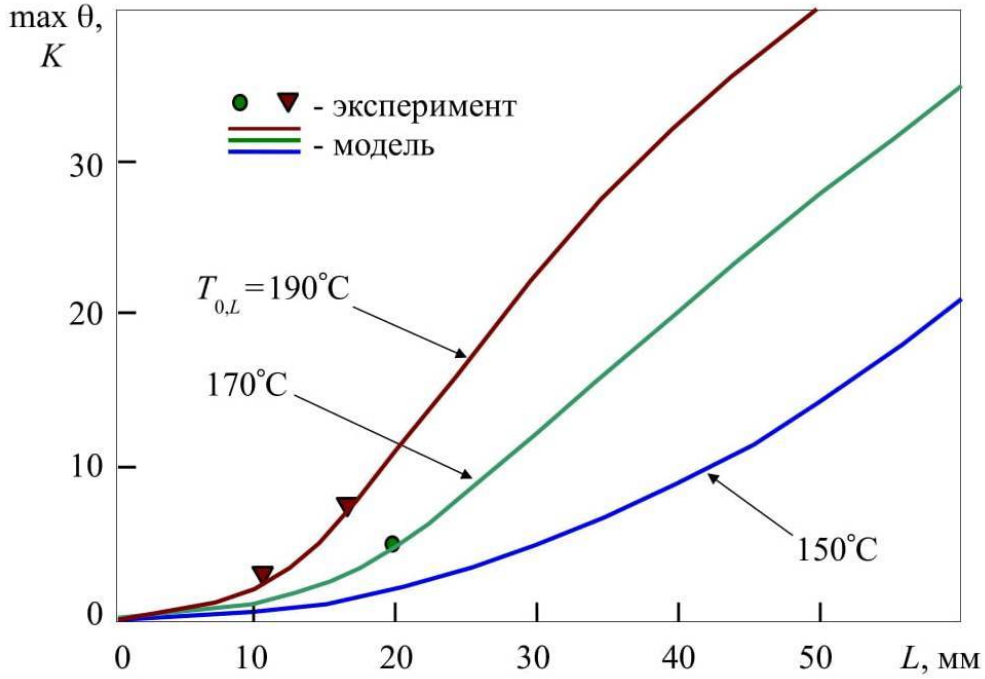


Рис. 1. Экзотермическая температура θ в зависимости от толщины L детали из углепластика при отверждении

Для процесса отверждения наиболее важным критерием оптимальности является минимизация времени полного отверждения изделия из ПК $t_к$ или минимизация напряжений σ, возникших из-за неравномерного поля отверждения. Тогда в результате решения задачи будут найдены такие режимные параметры изменения температуры во времени поверхностей деталей $U(t;\min t_к)=\{T_0(t),T_L(t)\}$ или $U(t;\min\sigma)=\{T_0(t),T_L(t)\}$, при которой целевая функция будет минимальна при использовании математических моделей [3, 6], описывающих ту или иную технологию производства композитных деталей.

Тогда профиль температурно-временного режима будет описываться линейными функциями на различных участках: нагреве, изотермической выдержке и их повторах, равных числу ступенек повышения температуры $k_{ст}$.

Задачу оптимизации дополняют ограничения, наложенные на решение. Это предельная температура $T_{\max}$, при превышении которой, связующее термически разлагается и изделие расслаивается или трескается. Следующие ограничения связаны с перепадом температуры внутри и на поверхности плоского изделия $\overline{\theta}$ и градиентом температуры $\overline{\chi}$ выражающие неоднородность температурного поля. Их превышение приводит к короблению изделия и созданию напряженной структуры композита. Численные значения этих ограничений определяются на основе специальных экспериментов моделирующих отверждение пластин с искусственно вызванными неоднородностями и последующими механическими испытаниями свойств ПК.

Используя исследованные характеристики ПК [2-4], с помощью программного обеспечения ИИС выполнены модельные расчеты и оценена возможность применения регламентного режима, предназначенного для отверждения тонких порядка 5 мм пластин, при производстве плоских деталей толщиной до 30 мм и более.

Анализ регламентного режима выполняли по результатам компьютерного моделирования температурно-конверсионных полей при отверждении плоских деталей толщиной L=30 мм. Моделировали одноступенчатый режим без ограничений с температурой отвер-

ждения 190°С и регистрацией температур плоскостей изделия T_0 и средних слоев $T_{L/2}$. В результате перегрев достиг $\theta = 22$ К ($T = 212$ °С), а градиент температуры χ = 2,8 К/мм, что является недопустимым по причине резкого ухудшения прочностных характеристик ПК из-за разложения связующего и растрескивания материала. Таким образом, регламентный режим может быть успешно применен только для изготовления пластин из ПК до 5 мм и приведет к браку при использовании его для более толстых деталей.

В случае ужесточения ограничений, налагаемых на решение задачи оптимизации, т.е. уменьшении их величины, число ступеней в режиме отверждения увеличивается с 2 до 5, и время полного отверждения также увеличивается. Соответственно уменьшаются температуры первых изотермических выдержек, а их продолжительность увеличивается. Анализ моделирования режимов отверждения плоских деталей из углепластика толщиной 30 мм, рассчитанных при задании различных ограничений, представлен на рисунке 2.

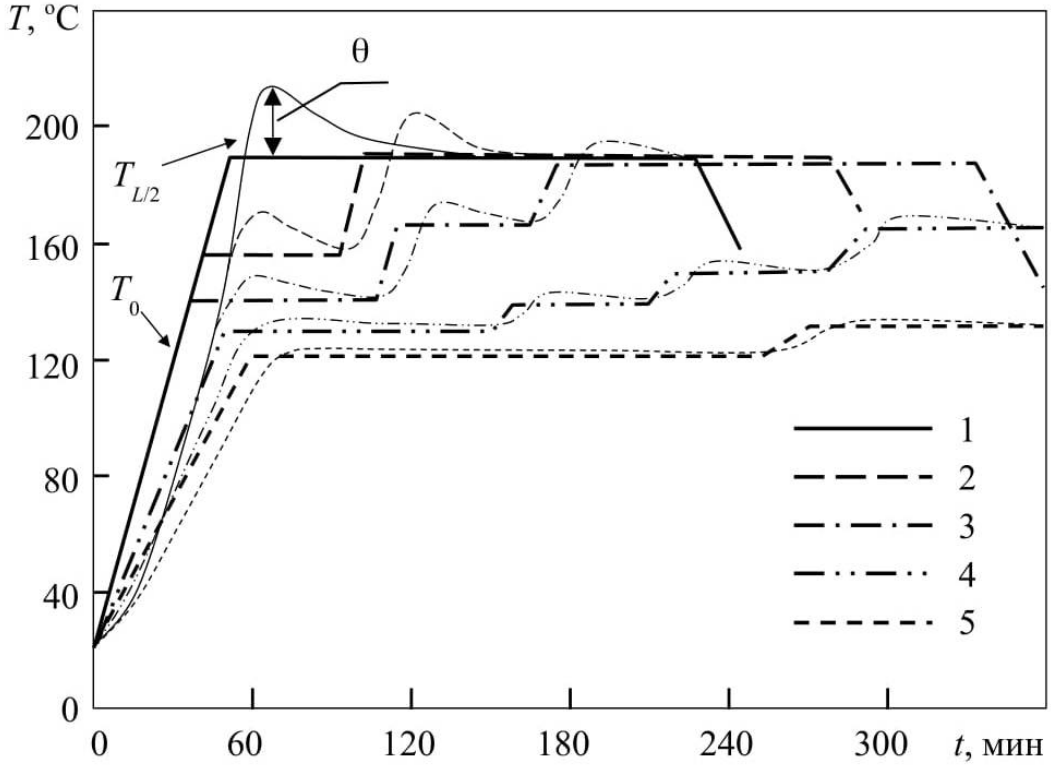


Рис. 2. Режимы отверждения деталей из углепластика толщиной L = 30 мм, рассчитанные при задании различных ограничений:

1 – регламентный режим, без ограничений θ = 22 К, χ = 2,8 К/мм;

2 – при задании ограничений $\overline{\theta}$ = 15 К, $\overline{\chi}$ = 2 К/мм;

3 – при задании ограничений $\overline{\theta}$ = 10 К, $\overline{\chi}$ = 1 К/мм;

4 – при задании ограничений $\overline{\theta}$ = 5 К, $\overline{\chi}$ = 0,4 К/мм;

(показаны четыре ступени режима, изображенного на рис. 3)

5 – оптимальный, при задании ограничений $\overline{\theta}$ = 5 К, $\overline{\chi}$ = 0,2 К/мм,

(показаны две ступени режима, изображенного на рис. 3)

Заключительной стадией является расчет оптимальных режимов отверждения исследованного углепластика. Рассчитанные оптимальные режимы отверждения минимальные по продолжительности показаны на рисунке 3, а также в таблице 1 представлены режимы, обеспечивающие минимальные внутренние напряжения. При расчете режимов на решение задачи

оптимизации накладывали ограничения: максимальный градиент температуры $\overline{\chi}$ = 0,2 К/мм и максимальный перепад температуры по толщине $\overline{\theta}$ = 5К.

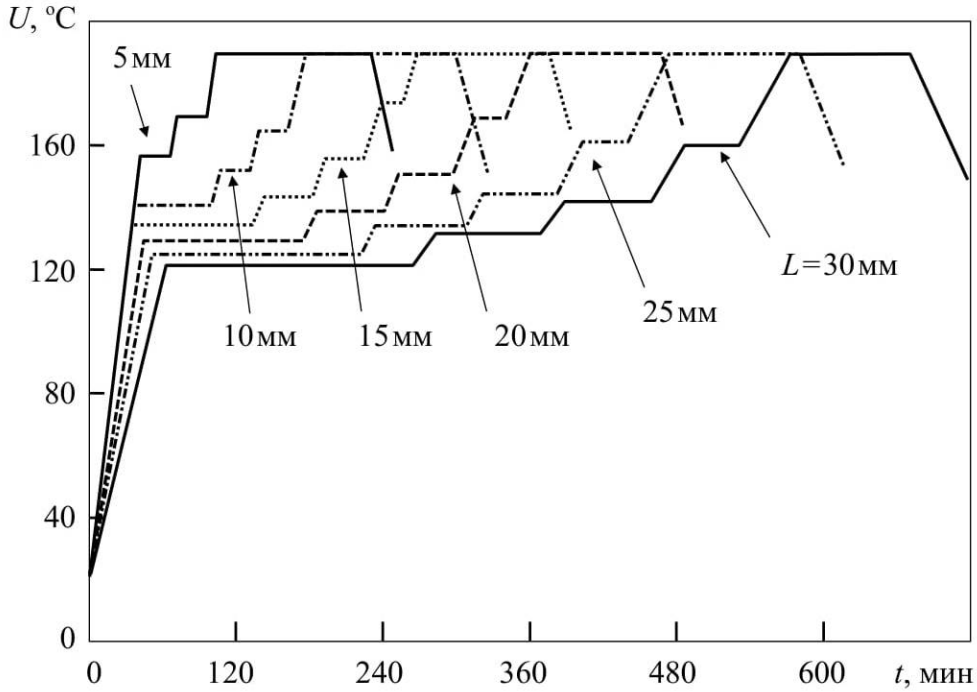


Рис. 3. Оптимальные режимы отверждения деталей
из углепластика минимальные по продолжительности

Таблица 1. Оптимальные режимы отверждения деталей из углепластика обеспечивающие минимальные внутренние напряжения

Толщина детали L, мм	Ступень нагрева	Параметры режима отверждения				
		Темп нагрева $\widetilde{K}_i$, К/мин	Время нагрева $t_{i\text{н}}$, мин	Температура выдержки $\widetilde{T}_i$, °С	Время выдержки $t_{i\text{выд}}$, мин	Время отверждения $t_{\text{к}}$, мин
5	1	3,0	45,5	156,3	29,6	260
	2	1,8	7,6	170,0	29,4	
	3	1,8	11,1	190,0	136,8	
10	1	3,0	40,3	141,0	69,7	340
	2	1,1	12,4	154,6	29,6	
	3	1,1	14,2	170,2	29,8	
	4	1,1	18,0	190,0	126,0	
20	1	2,4	45,1	128,5	145,4	528
	2	0,6	16,9	138,6	63,4	
	3	0,6	24,6	153,3	44,4	
	4	0,6	27,8	170,2	26,2	
	5	0,6	33,3	190,0	100,9	
30	1	1,7	60,6	121,4	221,4	744
	2	0,4	27,2	132,3	96,8	
	3	0,4	26,0	145,3	71,9	
	4	0,4	51,8	166,0	40,2	
	5	0,4	60,0	190,0	88,1	

Рассчитанные режимы в отличие от регламентного имеют значительно меньшие перепады θ и градиенты χ температуры. Из графиков (рис. 3) видно, что при увеличении толщины стенки плоских деталей из ПК количество ступенек увеличивается, температуры первых изотермических выдержек понижаются, а их продолжительность увеличивается, что способствует получению более однородной ненапряженной структуры ПК и как следствие качество материала и прочностные характеристики повышаются.

Таким образом, основным фактором, приводящим к перегреву внутренних слоев толстостенных деталей из полимерного композита на основе термореактивного связующего, является химическая реакция процесса отверждения, сопровождаемая интенсивным экзотермическим эффектом. Этот процесс приводит к деструкции связующего и созданию напряженной структуры материала. Поэтому необходима предварительная разработка оптимального режима отверждения и четкое его следование при изготовлении деталей из ПК. Особенно это важно при производстве толстостенных деталей.

В работе представлены результаты исследования влияния экзотермических процессов на режим отверждения и показаны режимы отверждения плоских деталей из углепластика различной толщины от 5 мм до 30 мм, которые были получены с помощью ИИС процесса отверждения, используя компьютерное моделирование и оптимизацию. Кроме того, компьютерное моделирование процесса отверждения позволяет просчитывать различные варианты ведения процесса и проводить анализ процесса с оценкой преимуществ и недостатков полученного режима для данного материала и выбирать правильную стратегию поиска оптимального режима. Это позволяет создавать изделия из полимерных композитов с максимально возможными прочностными характеристиками и с минимальной стоимостью, обеспечив максимальную производительность работы оборудования.

Библиографический список

1. Каблов, Е.Н. Материалы нового поколения и цифровые технологии их переработки // Вестник Российской академии наук. - 2020. Т. 90. № 4. - С. 331-334.

2. Dmitriev, O.S. Thermo-chemical analysis of the cure process of thick polymer composite structures for industrial applications / O.S. Dmitriev, A.A. Zhyvenkova, A.O. Dmitriev // Advanced Materials and Technologies. - 2016. № 2. - P. 53-60.

3. Дмитриев, О.С. Влияние режимов термообработки на геометрические и механические характеристики углепластиковых трубчатых элементов / О.С. Дмитриев, И.В. Малков // Вестник Тамбовского государственного технического университета. - 2016. Т. 22. № 3. - С. 427-438.

4. Computer-measuring system for research into properties of glutinous prepregs and calculation of curing cycles of the polymer composite materials on their base / O.S. Dmitriev, S.V. Mischenko, A.O. Dmitriev, V.N. Kirillov // Polymer Science, Series D. - 2010. V. 3. № 1. - P. 20-25.

5. Дмитриев, О.С. Устойчивый алгоритм для определения теплофизических характеристик / О.С. Дмитриев, А.А. Живенкова, А.А. Барсуков // Современные технологии в науке и образовании - СТНО-2021. сборник трудов IV Международного научно-технического форума. Рязань. - 2021. - С. 38-43.

6. Мищенко, С.В. Математическое моделирование процесса отверждения изделия из полимерных композиционных материалов методом горячего прессования / С.В. Мищенко, О.С. Дмитриев, С.В. Пономарев // Вестник ТГТУ. - 1998. – Т. 4. № 4. - С. 390-399.

УДК 66.067.55/.57; ГРНТИ 61.13

МОДЕРНИЗАЦИЯ ТЕХНОЛОГИИ ПОЛУЧЕНИЯ КРАСИТЕЛЯ ЧЕРНОГО ОРГАНИЧЕСКОГО ПРОИСХОЖДЕНИЯ

К.В. Брянкин, А.К. Брянкина
Тамбовский государственный технический университет,
Россия, Тамбов, nach_umu@tstu.ru

Аннотация. Изучен процесс синтеза и выделения твердой фазы в производстве парафенилдиамина восстановлением пара-нитроанилина (ПНА) до пара-фенилдиамина (ПФД) с использованием чугунной стружки. Определена оптимальная температура процесса восстановления ПФД. Предложен способ интенсификации процесса выделения целевого продукта из суспензии. Исследованы конвективный и кондуктивный методы сушки продукта. Оценено влияние температурного режима на эффективность интенсивность процесса сушки. Приведены результаты исследований различных методов удаления примесей из пасты Р-соли.
Ключевые слова: пара-фенилендиамин, краситель черный, чугунная стружка, реакция восстановления, термическая сушка, термическая чувствительность.

MODERNIZATION OF TECHNOLOGY FOR PRODUCING BLACK DYE OF ORGANIC ORIGIN

K.V. Bryankin, A.K. Bryankina
Tambov State Technical University,
Russia, Tambov, nach_umu@tstu.ru

The summary. The process of synthesis and isolation of the solid phase in the production of paraphenyldiamine by the reduction of para-nitroaniline (PNA) to para-phenyldiamine (PPD) using iron shavings has been studied. The optimal temperature for the PPD reduction process has been determined. A method is proposed for intensifying the process of isolating the target product from a suspension. The convective and conductive methods of drying the product have been studied. The influence of the temperature regime on the effectiveness of the drying process intensity is estimated. The results of studies of various methods for removing impurities from P-salt paste are presented.
Keywords: para-phenylenediamine, black dye, iron shavings, reduction reaction, thermal drying, thermal sensitivity.

На настоящий момент органические красители и пигменты занимают лидирующее положение в химической промышленности, так как в современном мире наблюдается значительная потребность в этом продукте. Особе значение данный факт приобретает в условиях жесткой санкционной политики в отношении нашей страны.

Пара-фенилендиамин (ПФД) представляет собой краситель черного цвета для окрашивания меховых изделий. Как и для любого красителя, так и для ПФД основными являются такие характеристики, как: дисперсный состав, цветность, чистота. Технология получения ПФД, в действующем производстве, заключается в восстановлении пара-нитроанилина (ПНА) до пара-фенилдиамина (ПФД), очистной фильтрации получаемого раствора ПФД от шлама, выделении ПФД в кристаллическом виде, разделении суспензии ПФД, сушки красителя и его упаковки.

Определяющей стадией в технологии получения ПФД является синтез основного вещества. Степень превращения ПНА в действующем производстве не превышает 70-75 %. Раствор ПФД, по окончании процесса, содержит большое количество смол (10-15 %), формирование которых объясняется несовершенством организации процесса перемешивания и малой эффективностью теплообмена.

Восстановление ПНА осуществляется путем перевода группы NO_2 - пара-нитроанилина в группу NH_2 - пара-фенилендиамина. В качестве восстанавливающего элемента используется железо, однако применение чистого железа нецелесообразно (дорого). Поэтому в промышленном производстве используют стальную или чугунную стружку, которая является отходом при механической обработки металлов. Стружка, имея специфическую

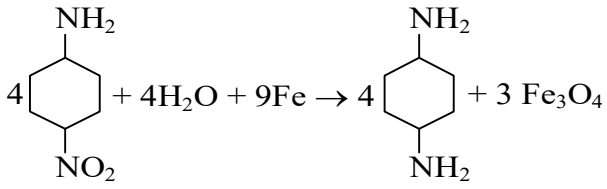
конфигурацию обеспечивает развитую контактную поверхность. Проблема с применением стружки, в частности - серого чугуна, заключается в сложности определения концентрации основного вещества (железа) в общей массе. Кроме того, чугунная стружка содержит большое количество окиси железа (5-7 %), углерода (в зависимости от сортности чугуна до 30 %) и загрязнений (до 5 %), а также в небольших количествах соединения других элементов: марганца (Mn), фосфора (P), кремния (Si), серы (S). Такая химическая неоднородность чугуна является причиной нестабильности выхода целевого продукта на стадии окисления.

Для обеспечения максимальной скорости процесса восстановления ПНА до ПФД необходимо наличие электролита ($FeCl_2$), поскольку его присутствие в воде усиливает коррозию чугунной стружки и увеличивает проводимость среды. Количество электролита должно быть 0,1-0,2 моль на 1 моль нитросоединения, что требует дополнительно чистого железа (62 кг на 1 тонну ПНА).

Когда содержание железа недостаточно для обеспечения максимального выхода по стадии восстановления, то часть исходного сырья - пара-нитроанилина, остается не прореагировавшей, и выход по стадии не превышает 75 %.

В механических цехах химических производств в основном имеются отходы серого чугуна марки СЧ-32, поэтому для синтеза ПФД со степенью превращения ПНА не менее 95 % был производен расчет расходных норм сырья, в том числе чугуна с учетом его марки.

Для взаимодействия железа с пара-нитроанилином:



,

и образования электролита:

$$2HCl + Fe \rightarrow FeCl_2 + H_2\uparrow ,$$

расход чугунной стружки (на 1 тонну ПНА) составил 1375 кг (с содержанием железа ≈ 70 %).

С учетом того, что в условиях производства соляная кислота имеет концентрацию ≈ 36 %, ее расход на образование электролита составил 223 кг (на 1 тонну ПНА).

Для гашения избыточного электролита по окончании процесса восстановления предлагается использовать кальцинированную соду:

$$FeCl_2 + Na_2CO_3 \rightarrow FeCO_3\downarrow + 2NaCl$$

в количестве 117 кг (на 1 тонну ПНА).

Описанный способ получения ПФД с использованием стружки чугуна марки СЧ32 не требует большого количества воды (до 4 $м^3$) и обеспечивает снижение потерь целевого вещества в растворенном виде по сравнению с действующим на настоящий момент промышленным производством с 50-70 г/л до 30-35 г/л.

Реакция восстановления ПНА до ПФД является экзотермической и сопровождается повышением температуры реакционной смеси до 110-150 °С, при этом активизируются побочные процессы окисления ПФД, и происходит значительное снижение содержания основного вещества до 60-65 %. Поэтому синтез ПФД необходимо проводить при температуре, не превышающей температуру разложения вещества. В результаты экспериментальных исследований было определено, что это соответствует 80 °С.

В промышленном производстве реакционная смесь с большим содержанием шлама и основного вещества в растворе поступает со стадии восстановления ПФД на очистную фильтрацию, которую осуществляют на фильтровальном оборудовании (фильтр - прессах) при температуре 90-100 °С. Промывку шлама осуществляют горячей водой при температуре 80-90 °С. Полученные промывные воды с концентрацией целевого компонента в растворенном виде 5-10 % возвращаются на стадию выделения. Для исключения процесса осмоления (окисления) ПФД и его потерь с промывными водами процесс фильтрования необходимо проводить при температуре 70-75 °С. Отмывку шлама осуществлять оборотной водой со стадии кристаллизации с подогревом до рабочей температуры 70-75 °С.

Процесс кристаллизации ПФД в существующем производстве проводится изогидрическим методом в емкостных аппаратах с мешалками и теплообменными устройствами охлаждением раствора ПФД с 90 °С до 12 °С со скоростью 20 °С/час и выдержкой в течении 20 часов, при этом получаются кристаллы игольчатой формы с эквивалентным диаметром около 0,5 мм. Непрерывное охлаждение раствора наряду с ростом кристаллов обеспечивает постоянное возникновение новых центров кристаллизации [1], что приводит к получению продукта с широким диапазоном по гранулометрическому составу. Необходимо было определить такие технологические режимы процесса кристаллизации, при которых обеспечивается максимальный выход продукта и увеличиваются размеры и однородность получаемых кристаллов.

Решить эти проблемы удалось за счет проведения кристаллизации со ступенчатой организацией охлаждения раствора ПФД и добавлением затравки [2].

Введение затравки в виде суспензии мелкодисперсных частиц ПФД с концентрацией 20-25 % и при температуре 8 °С приводит к быстрому образованию и росту кристаллов.

Контроль и регулировка температурного режима процесса кристаллизации позволила получить однородный дисперсный состав кристаллов с заданным эквивалентным диаметром.

Исследования процесса кристаллизации проводились на установке, принципиальная схема которой показана на рисунке 1. Модель кристаллизатора представляет собой емкостной аппарат объемом 500 мл, оснащенный охлаждающей рубашкой и перемешивающим устройством. Охлаждение осуществлялось рассолом.

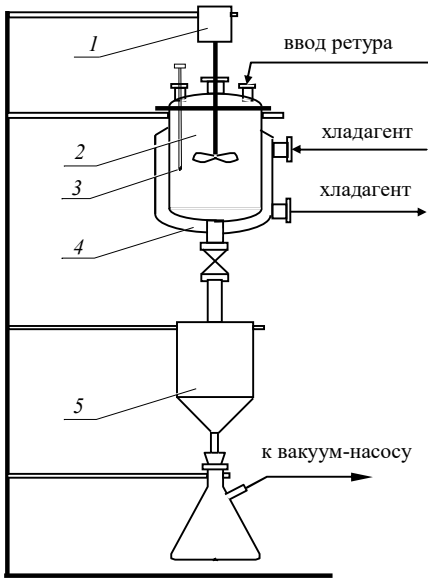


Рис. 1. Схема лабораторного стенда для исследования процесса кристаллизации:
1 – перемешивающее устройство; *2* – модель кристаллизатора; *3* – термопара;
4 – рубашка; *5* – модель вакуум-фильтра

Исходный раствор ПФД с концентрацией по целевому веществу 350 г/л при температуре 80 °С помещался в кристаллизатор и охлаждался до температуры 32 °С, соответствующей началу кристаллообразования, затем в кристаллизатор добавлялась затравка в количестве 1 % от объема раствора и проводилось дальнейшее охлаждение раствора со скоростью 4-5 °С/час при работающей мешалке. Интенсивность перемешивания должна быть достаточной для устранения локальных зон переохлаждения у стенок аппарата и исключения возможности возникновения новых центров кристаллизации [2]. Охлаждение проводили до температуры 25 °С и делали выдержку в течении времени необходимого для наступления состояния насыщения (около 5 часов). Этому периоду соответствует рост образовавшихся кристаллов и формирование новых центров кристаллизации. Температурный режим следующего периода характеризуется медленным охлаждением раствора до температуры 12-15 °С при работающей мешалке, скорость охлаждения около 10 °С/час. Этому периоду соответствует рост кристаллов до величины 1,5-2,0 мм. При достижении температуры 12 °С проводится быстрое охлаждение до 8 °С и выдержка для полного выделения ПФД из раствора. В целом, организация температурного режима кристаллизации ПФД из раствора схематически показана на рисунке 2.

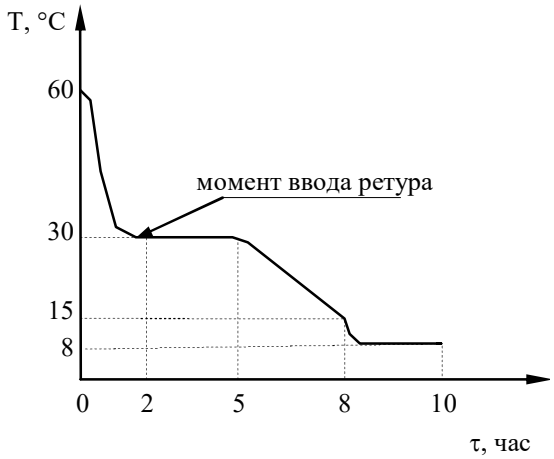


Рис. 2. Температурный режим кристаллизации ПФД

После завершении процесса кристаллизации проводился анализ пасты и отработанного маточного раствора на содержание ПФД. Проведение кристаллизации по вышеописанной схеме позволяет получить продукт с концентрацией ПФД в пасте около 78-80 %, а потери основного вещества с маточным раствором снизить до 30 г/л.

Анализ полученных данных показывает, что использование ретура и ступенчатого режима охлаждения раствора обеспечивает сокращение длительности процесса в 2 раза при уменьшении потерь продукта на 20-25 % и увеличение размеров кристаллов с 0,5 до 1,5-2 мм. Предлагаемый ступенчатый режим кристаллизации может быть рекомендован для аналогичных производств синтеза органических красителей.

После выделения из суспензии ПФД в виде пасты с концентрацией 60-65 % на существующем производстве проводят сушку ПФД в вакуум-гребковой сушилке (типа «Венулет») при температуре 45-50 °С. Длительность стадии достигает 4 суток, максимальная концентрация целевого продукта на выходе не превышает 92 %. Необходимо высушить продукт до концентрации 99,5 %.

Для сушки ПФД (ввиду его термолабильных свойств) можно предложить методы с интенсивным режимом удаления влаги (в виброаэрокипящем слое) [3, 4].

Исследование процесса сушки ПФД в виброаэрокипящем слое материала проводилось на лабораторной модели сушилки (рис. 3). В качестве объекта исследования использо-

валась паста ПФД с исходной концентрацией 52÷55 %. Сушильным агентом являлся воздух с температурой 25-30 °С.

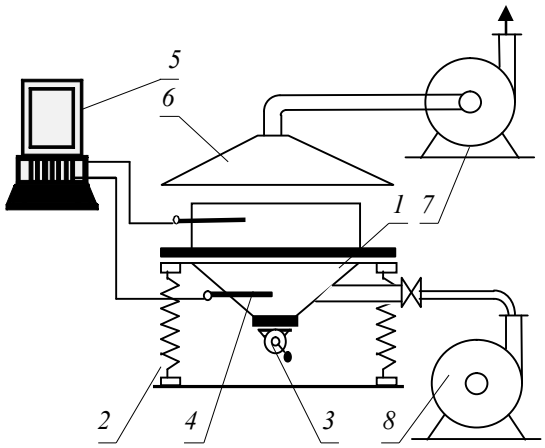


Рис. 3. Схема экспериментальной установки по изучению процесса сушки в виброкипящем слое:
1 – корпус сушилки; *2* – пружинный амортизатор; *3* – вибратор;
4 – термопары; *5* – контрольный самопишущий прибор; *6* – система вытяжки;
7 – вытяжной вентилятор; *8* – воздуходувка

Эксперимент проводился следующим образом: устанавливался угол наклона лотка таким образом, чтобы в рабочем состоянии отсутствовало продольное перемещение материала вдоль лотка и обеспечивалось равномерное распределение материала по его длине; сушилка прогревалась горячим воздухом в течение 20-30 минут; паста равномерно распределялась по лотку толщиной 1,5-2 см; процесс сушки проводился в течении 4-5 мин; сухой продукт анализировался на содержание целевого вещества и остаточную влажность.

Необходимая концентрация ПФД (98,5-99,5 %) достигалась в течение 30 мин.

Выводы

1. Оптимальная температура процесса восстановления ПФД соответствует 75-80 °С.
2. использование чугунной стружки с учетом ее составляющих и уточненных расходных норм сырья и полупродуктов позволяет сократить потери ПФД в растворенном виде с 50-60 г/л до 30-35 г/л и снизить расход воды вдвое (до 4 м^3).
3. Очистная фильтрация при температуре 70-75 °С снижает потери целевого продукта на 10-15 %.
4. Выделение ПФД в кристаллическом виде с использованием ступенчатого режима охлаждения и затравки обеспечивает сокращение длительности процесса в 2 раза, сокращению потерь ПФД до 30 г/л, увеличению размеров кристаллов до 1,5-2,0 мм.
5. Сушку продукта рекомендуется проводить в виброаэрокипящем слое при температуре сушильного агента (воздуха) 25-30 °С.

Библиографический список:

1. Гельперин Н.И., Носов Г.А. Основы техники фракционной кристаллизации. – М.: Химия, 1986. – 303 с.
2. Лебеденко Ю.П. Кристаллизация из растворов в химической промышленности. – Л.: Химия, 1973. – 48 с.
3. Брянкин, К.В. Исследование термодеструктивных превращений, протекающих при температурном воздействии на полупродукты органических красителей / К.В. Брянкин, А.А. Дегтярев // Казанская наука. 2010. № 1. – С. 14-18.
4. Брянкин К.В. Классификация и выбор сушилок для полупродуктов органических красителей с учетом их термоустойчивости / К.В. Брянкин, А.И. Леонтьева // Фундаментальные исследования. – 2011. – № 8. Ч. 1. – С. 116-119.

УДК 621.357.8; ГРНТИ 55.22

ОСОБЕННОСТИ ТРАВЛЕНИЯ ВО ФТОРСОДЕРЖАЩИХ СРЕДАХ

Р.В. Бекташев*, Е.В. Воробьева**
**Рязанский государственный радиотехнический университет имени В.Ф. Уткина,*
Российская Федерация, Рязань, bektashev99@mail.ru,
***Рязанский институт (филиал) Московского политехнического университета*
Российская Федерация, Рязань, vorobeva_70@bk.ru

Аннотация. В работе рассматриваются процесс подготовки поверхности титана и составы фторсодержащих травителей, приводится анализ растворов травления.
Ключевые слова: фторсодержащие среды, травление титана, гравиметрический метод

SPECIFICITIES OF ETCHING IN FLUORINATED SUBSTANCES

R.V. Bektashev*, E.V. Vorobyova**
**Ryazan State Radio Engineering University named after V.F. Utkin,*
Russia, Ryazan, bektashev99@mail.ru,
***Ryazan Institute (branch) Moscow Polytechnic University*
Russian Federation, Ryazan, vorobeva_70@bk.ru

Annotation. The paper discusses the process of preparing the titanium surface and the compositions of fluorinated etchants.
Keywords: fluorinated substances, etching of titan.

Титан - тугоплавкий металл с невысокой плотностью. Удельная прочность титана выше, чем у многих легированных конструкционных сталей, поэтому при замене сталей титановыми сплавами можно при равной прочности уменьшить массу детали на 40% [1].

За счет своей реакционной стойкости титан применяют в химической промышленности в качестве материала для трубопроводов и аппаратов.

Также за счет своей повышенной холодостойкостью и устойчивостью к высоким температурам титан применяется в качестве конструкционного материала в космической промышленности и ракетостроении.

В авиапромышленности титан ценен за счет своей низкой плотности. При одной и той же прочности сплавы титана весят гораздо меньше стальных сплавов.

Помимо этого, титан нашел себе применение и в медицине за счет своей инертности, легкости и прочности.

Часто на поверхность титана наносят покрытия на основе оксидов и оксинитридов титана. Данные покрытия придают коррозионную стойкость сплавам титана.

Перед нанесением данных покрытий, поверхность титана необходимо подготовить. Основной проблемой подготовки поверхности титана является применение травильных растворов на основе плавиковой кислоты. Данная кислота обладает повышенной токсичностью и представляет огромную опасность для человека [2].

Целью данной работы является изучение процесса подготовки поверхности титана и сравнение некоторых составов травильных растворов на основе минеральных кислот в смеси с фторидом аммония.

Широкое применение сплавов титана и необходимость подготовки поверхности титана перед нанесением защитных антикоррозионных покрытий еще раз подчеркивают актуальность данного исследования.

В соответствии с поставленной целью необходимо решить следующие задачи:

- изучить процесс подготовки поверхности титана;
- составить растворы травления нескольких разных составов и произвести травление образцов;
- произвести гравиметрический анализ и на его основе сравнить растворы травления.

Подготовка поверхности включает в себя 4 этапа: шлифование, полирование, обезжиривание и травление.

Шлифование и полирование помогают убрать микровыступы и неровности поверхности. Обезжиривание производят для очистки поверхности металла от загрязнений. Чаще всего в качестве обезжиривателей используют щелочные растворы, либо в случае титана четыреххлористый углерод.

Обезжиривание позволяют удалить лишь загрязнения, внесенные извне. Ржавчина, окалина и окисные пленки удаляются химически с помощью травления.

Травление производят во фторсодержащих средах. Применяют растворы плавиковой кислоты в смеси с другими кислотами, либо растворы минеральных кислот в смеси с солями фтора. Чаще всего применяют фторид аммония [3].

Фторсодержащие добавки необходимы для препятствия процесса наводораживания поверхности металла.

В ходе исследования данного вопроса нами был смоделирован и произведен эксперимент. Для сравнения использовались образцы размером 5х5х0,1 см. Сначала производилось обезжиривание образцов в растворе четыреххлористого углерода. Далее осуществлялась промывка образцов дистиллированной водой, с последующим высушиванием в сушильном шкафу при температуре 120 °С. Далее производился замер массы образцов до травления.

После замера снова производилось обезжиривание образцов в четыреххлористом углероде с дальнейшей промывкой образцов в дистиллированной воде.

Следующим этапом было приготовление травильных растворов. В качестве травильных растворов использовались соляная, серная, азотная и фосфорная кислоты в концентрации 7 моль/л с добавлением соли фторида аммония в концентрации 0,35 моль/л. Каждый из травильных растворов приготовлен в объеме равном одному литру.

Травильный раствор с HCl готовился из 600 мл 36% HCl, 13 грамм NH_4F и дистиллированной воды.

Травильный раствор с H_2SO_4 готовился из 374 мл 98% H_2SO_4, 13 грамм NH_4F и дистиллированной воды.

Травильный раствор с HNO_3 готовился из 610 мл 54% HNO_3, 13 грамм NH_4F и дистиллированной воды.

Травильный раствор с H_3PO_4 готовился из 480 мл 85% H_3PO_4, 13 грамм NH_4F и дистиллированной воды.

Процесс травления занял 45 минут. После травления все образцы были промыты дистиллированной водой и высушены в сушильном шкафу. Затем производилось измерение массы образцов после травления.

Далее анализировали полученные данные и проводили сравнение травильных растворов.

Сравнение травильных растворов производили с помощью гравиметрического метода по таким характеристикам, как степень травления, скорость травления и высота стравленного слоя.

Степень травления – характеристика процесса травления, которая показывает изменение массы при травлении образца и определяется по формуле:

$$a = \frac{m_0 - m}{S}, \quad (1)$$

где a – степень травления, г/см^2;

m_0 – масса образца до травления, г;

m – масса образца после травления, г;

S – площадь поверхности образца, см^2.

Скорость травления – характеристика процесса травления, которая показывает изменение толщины пластины и определяется по формуле:

$$\omega = \frac{m_0 - m}{S \cdot \rho \cdot t} \cdot 10000, \quad (2)$$

где ω – скорость травления, мкм/мин;
m_0 – масса образца до травления, г;
m – масса образца после травления, г;
S – площадь поверхности образца, $см^2$;
ρ – плотность образца, $г/см^3$;
t – время травления, с.

Высота стравленного слоя определяется по формуле:

$$h = \frac{m_0 - m}{S \cdot \rho} \cdot 10000, \quad (3)$$

где h – высота стравленного слоя, мкм;
m_0 – масса образца до травления, г;
m – масса образца после травления, г;
S – площадь поверхности образца, $см^2$;
ρ – плотность образца, $кг/см^3$.

Результаты гравиметрического анализа представлены в таблице:

Таблица 1 – Сравнение травильных растворов

	H_3PO_4:NH_4F	H_2SO_4:NH_4F	HNO_3:NH_4F	HCl:NH_4F
Масса образца до травления, г	11,3512	11,3485	11,3508	11,3479
Масса образца после травления, г	11,3144	10,8102	10,7844	10,7702
Степень травления, $г/см^2$	0,001472	0,021532	0,022656	0,023108
Скорость травления, мкм/мин	0,072	1,054	1,109	1,131
Высота стравленного слоя, мкм	3,240	47,430	49,905	50,895

Анализ результатов, полученных в ходе исследования, позволяет сделать следующие выводы:

- раствор с ортофосфорной кислотой обладает малой скоростью травления, в то время как остальные растворы обладают достаточной скоростью травления;

- растворы с азотной и соляной кислотами парят и тем самым представляют опасность, так как возможен ожог дыхательных путей;

- оптимальным по безопасности и скорости травления является раствор с серной кислотой.

Библиографический список

1. Лебедев, В. А. Металлургия титана : учеб.пособие / В. А. Лебедев, Д. А. Рогожников – Екатеринбург : Издательство УМЦ УПИ, 2015. – 194 с..

2. Файнзильберг А.А. Фтористый водород как реагент и среда в химических реакциях. / А.А. Файнзильберг, Г.Г. Фурин – М.: Наука, 2008. – 312 с.

3. Усова В.В., Плотникова Т.Н., Кушакевич С.А. Травление титана и его сплавов. М. : Металлургия, 1984. – 128 с..

УДК 628.33; ГРНТИ 70.25.17

СОВЕРШЕНСТВОВАНИЕ МЕТОДОВ ОЧИСТКИ СТОЧНЫХ ВОД ГАЛЬВАНИЧЕСКОГО ПРОИЗВОДСТВА

А.В. Порхунова*, Е.Р. Корнешова*, Е.В. Воробьева**
**Рязанский государственный радиотехнический университет имени В.Ф. Уткина,*
Российская Федерация, Рязань, porhunovaalbina@yandex.ru
***Рязанский институт (филиал) Московского политехнического университета*
Российская Федерация, Рязань, vorobeva_70@bk.ru

Аннотация. Рассмотрены различные технологии совершенствования методов очистки сточных вод гальванического производства.
Ключевые слова: гальваническое производство, сточные воды, реагенты, биологическая очистка, биотенки, феррофферригидрозол, гетерокоагуляция

IMPROVEMENT OF WASTEWATER TREATMENT METHODS ELECTROPLATING PRODUCTION

A.V. Porkhunova*, E.R. Korneshova*, E.V. Vorobyova**
**Ryazan State Radio Engineering University named after V.F. Utkin,*
Russia, Ryazan, porhunovaalbina@yandex.ru
***Ryazan Institute (branch) Moscow Polytechnic University*
Russian Federation, Ryazan, vorobeva_70@bk.ru

The summary. Various technologies for improving the methods of wastewater treatment of galvanic production are considered.
Keywords: lectroplating production, waste water, reagents, biological purification, biotenes, ferroferrigidrozole, heterocoagulation

Гальваническое производство является одним из наиболее опасных с экологической точки зрения. В большом объеме промывных и сточных вод содержатся практически все ионы тяжелых металлов, неорганические кислоты и щелочи, поверхностно-активные реагенты, твердые высокотоксичные соединения. Технология гальванического производства должна быть взаимно адаптирована с процессом, в ходе которого сточные воды будут очищаться.

Методы очистки сточных вод гальванического производства разнообразны, но не все позволяют достичь выполнения требований: очистки до норм ПДК сточных вод, возврата воды на оборотное водоснабжение, извлечения ценных компонентов. Таким образом, необходимость исследования и поиск более эффективных методов очистки сточных вод от тяжелых металлов актуальна, имеет как научную, так и практическую значимость.

Целью исследования являются поиски более совершенствованных методов очистки сточных вод гальванического производства, подтверждение экологической безопасности и обоснование эффективности применения предлагаемых методов.

Одним из предлагаемых методов является биологическая очистка, которая возможна в естественных условиях и в искусственных сооружениях. И в том, и в другом случае органические примеси обрабатываются бактериями, простейшими, водорослями и превращаются в минеральные вещества.

В естественных условиях очистка производится на полях фильтрации или орошения, т.е. через почву, или в биологических прудах-отстойниках, в которых концентрация загрязнителей снижается до требуемых норм за счет процессов самоочищения, осуществляемых микроорганизмами, водорослями, беспозвоночными (потребления микроорганизмами органических составляющих сточных вод). Пруды могут быть с поддувом воздуха (с искусственной аэрацией). Для самоочищения водоемов используются высшие водные растения (тростник, камыш, уруть, ряска, тропическое цветковое растение – эйхорния или водный гиацинт) [2].

В качестве искусственных сооружений для промстоков применяются аэротенки, окситенки и биофильтры (аэро – с подачей воздуха; окси – с подачей кислорода). В тенках (железобетонных резервуарах) сточные воды обрабатываются микроорганизмами. Для их функционирования необходима определенная температура, кислотность (pH) и отсутствие многих солей (микроорганизмы могут погибнуть) [2].

В комплексе инженерных сооружений применяются биофильтры, в которых активная биологическая среда образуется на специальной загрузке (шлак, керамзит, гравий и т.п.). Эта биологическая среда (пленка) менее чувствительна к колебаниям параметров среды и сточных вод. Активность биопленки увеличивается при продувании воздухом, подаваемым противотоком.

Проведенные исследования показали. что эффективным методом очистки сточных вод гальванического производства является обезвреживание стоков с применением феррроферригидрозоля (ФФГ). Композиции ФФГ – коллоидная суспензия гидратированных соединений двух- и трехвалентного железа как реагента для связывания загрязнений в стоках. ФФГ производится раздельно в процессе электролиза из отходов железа.

Раздельное изготовление дает возможность подбирать особые условия процесса: проводить электролиз при более низком расходе электроэнергии, регулировать токопроводность, температуру и состав электролита для большего выхода по току гидратированных ионов железа, а также вводить добавки, необходимые для интенсификации процесса. Состав производится электрохимическим путем в специальных генераторах, входящих в комплект оборудования. Сырьем могут служить отходы штамповки, стальная стружка и т.п.

Во время электрохимического процесса в растворе происходит оксидация железа и образуются ионы железа, которые тут же гидролизуются. Химический процесс, происходящий в пространстве вокруг железного анода, диффузия ионов в раствор, гидролиз и образование продуктов гидролиза в виде нерастворимых наночастиц в результате гидролитической полимеризации. В растворе последовательно происходят следующие превращения:

$$Fe \rightarrow Fe^{2+} + 2e^{-}$$
$$Fe + OH^{-} \rightarrow FeOH^{+} + 2e^{-}$$
$$FeOH^{+} + OH^{-} \rightarrow Fe(OH)_2$$
$$Fe(OH)_2 \rightarrow Fe_nOH^{+} + OH^{-}$$
$$Fe(OH) \rightarrow Fe^{2+} + OH^{-}$$

Основной механизм осаждения токсичных веществ – гетерокоагуляция. ФФГ вступает в реакции с ионами, гидроксидами и мицеллами цинка, хрома и других тяжелых металлов в одном диапазоне pH. ФФГ является хорошим коагулянтом и сорбентом, восстановителем и химическим реактивом [1].

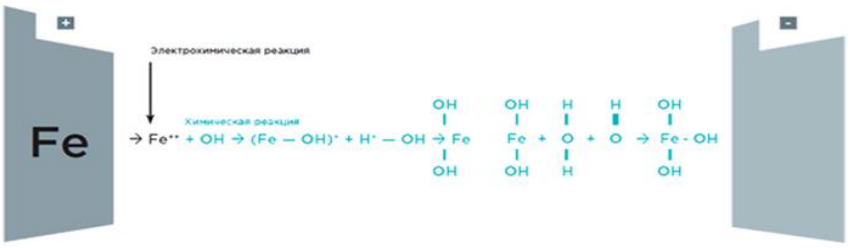


Рис. 1. Электрохимический процесс получения ФФГ

Содержащиеся в сточных водах гидратированные ионы меди, цинка, хрома, никеля, кадмия способны как сорбироваться ФФГ, так и вступать во взаимодействие с соединениями железа. Глубина очистки возрастает в результате образования смешанных кристаллов и химических соединений. Окислительно - восстановительные потенциалы реагента в водных растворах позволяют восстановить как бихромат- ионы, так и хромат-ионы. Высокая эффективность гидратообразования ионов тяжелых металлов обусловлена одновременным протеканием обменных и окислительно-восстановительных реакций.

ФФГ способен взаимодействовать и с катионами металлов, и с анионами, склонными к комплексообразованию, и с ионами металлов, связанных в комплексы. Благодаря этому, стоки очищают до ПДК даже при наличии в них сильных комплексов-пирофосфатов, тартратов, цитратов, аммонийных соединений.

Ферроферригидризоль не вызывает засоления стоков в ходе их обезвреживания. Этим облегчается возврат воды в основное и вспомогательное производство. Между тем при традиционной реагентной очистке повторное использование возможно лишь в случае дорогостоящей доочистки с помощью ионообменных смол.

Схема очистки сточных вод с использованием ферроферригидризоля существенно упрощена. Будучи эффективным восстановителем, реагент переводит содержащийся в отработанных водах шестивалентный хром в трехвалентный, который в щелочной среде выпадает в осадок. Поэтому обезвреживание производится без разделения стоков на хромосодержащие и кислотно-щелочные. Вместе с ионами тяжелых металлов удаляются смазочные-охлаждающие жидкости, остатки нефтепродуктов, красители, органические добавки, детергенты, уменьшаются ХПК и БПК.

Технологическая схема обезвреживания стоков имеет многие сходства с обычной реагентной схемой сточных вод. Суспензия ФФГ растворяется на аноде и является основным осаждающим реагентом. Из накопителя готовый реагент попадает в реактор, где смешивается с отработанными водами. Растворы щелочи и кислоты доводят pH до нужного значения, к тому же могут применяться и готовые растворы [1].

Гетерокоагуляция интенсифицирует процесс отстаивания взвесей. Он может осуществляться без отстойников – непосредственно в реакторе, как и обработка стоков без их разделения, это сокращает число единиц оборудования и площадь, необходимую для их размещения.

Обезвреживание стоков идет в непрерывном или периодическом режиме. Осадок (шлам), образующийся в результате очистки, поступает на фильтр-пресс и обезвоживается. Он малотоксичен и обогащен железом. Это делает шлам пригодным к вывозу на обычные полигоны для хранения отходов и ценным сырьем для производства пигментов, стройматериалов.

Таким образом, нами были исследованы различные технологии совершенствования методов очистки сточных вод гальванического производства.

Проведенный анализ позволяет сделать вывод, что самыми перспективными технология очистки сточных вод гальванического производства являются метод с использованием ферроферригидризоля и биологическая очистка. Итогом применения данных технологий может явиться планомерное уменьшение концентрации тяжелых металлов в очищенных сточных водах до показателей, колеблющихся в пределах следовых концентраций.

Библиографический список

1. Д.Будиловскис, П.Балтренас, Д.Щупакас, Л.С.Ещенко. Составы и свойства осадков, полученных при очистке сточных вод ферроферригидрозолем // Химическое и нефтегазовое машиностроение. – 2004, № 11.

2. Яковлев С.В., Карюхина Т.А. Биохимические процессы в очистке сточных вод. - М.: Стройиздат, 1990.

УДК 004.94; ГРНТИ 28.17.33

ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ РАБОТЫ РЕКТИФИКАЦИОННОЙ КОЛОННЫ ДЛЯ РАЗДЕЛЕНИЯ ДВУХКОМПОНЕНТНОЙ СМЕСИ

В.В. Коваленко, Н.Ю. Кулавина, М.В. Лызлова, Г.А. Шашкина
Рязанский государственный радиотехнический университет имени В.Ф. Уткина,
Россия, Рязань, xt-kontakt@mail.ru

Аннотация. В данной работе в программе CAPE-OPEN to CAPE-OPEN simulation моделируется ректификационная колонна по рассчитанным исходным данным.
Ключевые слова: бинарная смесь, уравнение состояния Пенга-Робинсона, модель ректификационной колонны.

STUDY OF THE OPERATION EFFICIENCY OF A RECTIFYING COLUMN FOR SEPARATING A TWO-COMPONENT MIXTURE

V.V. Kovalenko, N.U. Kulavina, M.V. Lyzlova, G.A. Shashkina,
Ryazan State Radio Engineering University named after V.F. Utkin,
Russia, Ryazan, xt-kontakt@mail.ru

Annotation. In this work, in the CAPE-OPEN to CAPE-OPEN simulation program, a distillation column is modeled according to the calculated initial data.
Keywords: binary mixture, Peng-Robinson equation of state, model of the distillation column.

Разделение бинарной смеси, состоящей из компонентов с разной температурой кипения, осуществляют в противоточных колонных аппаратах с различными контактными элементами, такими как насадки или тарелки. Расчет и проектирование ректификационных установок сводится к определению основных геометрических размеров ректификационной колоны – ее диаметра и высоты. В значительной степени эти размеры зависят от гидродинамического режима работы колонны, от скоростей и физических свойств фаз, от характеристик используемых контактных устройств. Для определения скоростей потоков необходимо знать нагрузки по пару и жидкости в колонне, которые определяются значением рабочего флегмового числа (R), а его оптимальное значение ($R_{опт}$) находится путем технико-экономического расчета. Однако в большинстве случаев для оценки ($R_{опт}$) используют приближенные вычисления, основанные на определении коэффициента избытка орошения. Оптимальное значение флегмового числа зависит от его минимальной величины и рассчитывается исходя из состава исходной смеси (питании) и дистиллята – мольных долей низкокипящего компонента. Таким образом, исследование эффективности работы ректификационной колонны можно проводить с использованием современных моделирующих программ, задаваясь различными значениями коэффициента избытка флегмы, определять расход по пару и жидкости и возможность получения дистиллята и остатка с заданными характеристиками при разделении исходной смеси в ректификационной колонне с определенными контактными устройствами и размерами (диаметром).

В качестве примера рассматривается процесс разделения бинарной смеси хлороформ-бензол в ректификационной установке непрерывного действия производительностью F (кг/ч) с начальной температурой $t_н$ °С в программе CAPE-OPEN to CAPE-OPEN simulation [1]. В результате аналитических расчетов получены исходные данные, необходимые для построения модели.

1. Мольные доли хлороформа в исходной смеси F-x_F, дистилляте D- x_D и кубовом остатке W- x_W.

2. Расходы исходной смеси, дистиллята и кубового остатка: F (кг/с), D (кг/с) и W(кг/с).

3. Оптимальное флегмовое число $R_{опт}$.

4. Число тарелок в колонне $N_т$.

5. Температуры исходной смеси, дистиллята и кубового остатка: t_F °С; t_D °С , t_W °С.

6. Температуры охлажденного дистиллята и кубового остатка $t_к$ °С.

7.Температура греющего пара $t_{гр.п}$ °С, температуры охлаждающей воды $t_в$ °С.

8. Температуры кипения компонентов: хлороформ $t_{хл}$ (61,5 °С), бензол $t_б$ (80,6 °С).

Модель ректификации бинарной смеси рассматривается в адиабатных условиях и допущении незначительного перепада давления в колонне, то есть теплопотери и перепад давления на каждой тарелке принимаются равными нулю [2]. Дополнительно приняты допущения:

- равенство давлений в конденсаторе, в кипятильнике и в колонне,
- равенство давлений исходной смеси и на тарелке питания,
- подача флегмы при температуре ее кипения.

Для входного потока задаются давление, температура, расход, мольные доли исходной смеси (рисунок 1) [3]. Из расчета потока смесь находится в жидкой фазе.

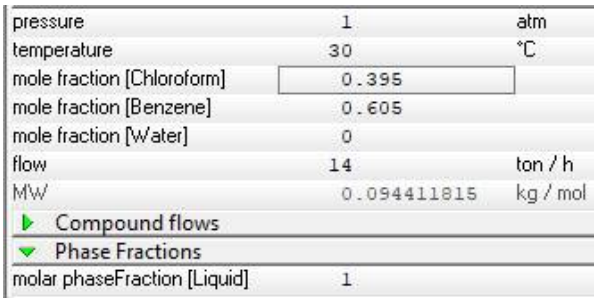
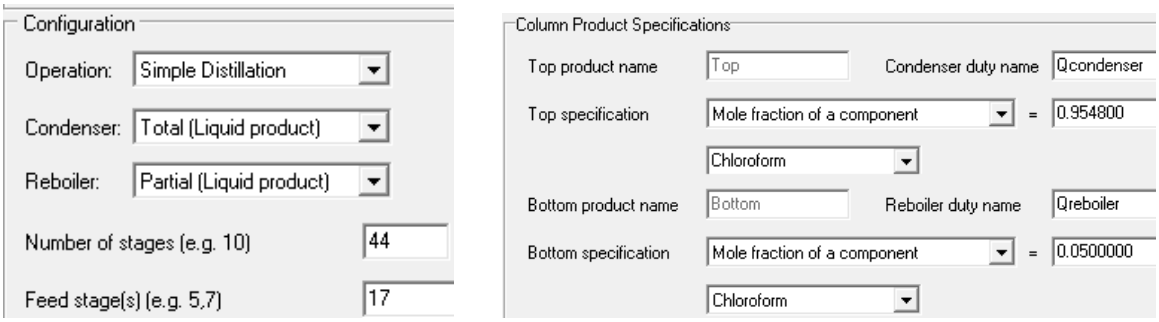


Рис. 1. Параметры входного потока

На входе в ректификационную колонну температура исходной смеси определяется по диаграмме t-x,y, которая равна 76 °С. Для модели теплообменника, применяемого для нагрева исходной смеси и кубового остатка, задается встречный поток пара с температурой $t_{гр.п}$ (133 °С) и на выходе теплообменника определяется заданная температура.

Для модели колонны задаются: простая дистилляция с конденсатором и ребойлером, количество тарелок $N_т$ (44), тарелка ввода сырья в колонну 17 (рисунок 2, а), уравнение состояния Пенга-Робинсона, давление постоянное равное 1 атм, эффективность тарелок 1, расчетная мольная доля хлороформа для верхнего x_F (0,9548) и для нижнего x_W (0,11) продуктов (рисунок 2, б).



а) б)

Рис. 2. Окна ввода: а) конфигурация колонны; б) параметры верха и низа колонны

Конденсация паров дистиллята осуществляется в холодильнике-конденсаторе водой с температурой $t_в$ °С.

Схема модели включает в себя также теплообменники для охлаждения дистиллята и остатка до 25 °С. Полностью схема модели приведена на рисунке 3.

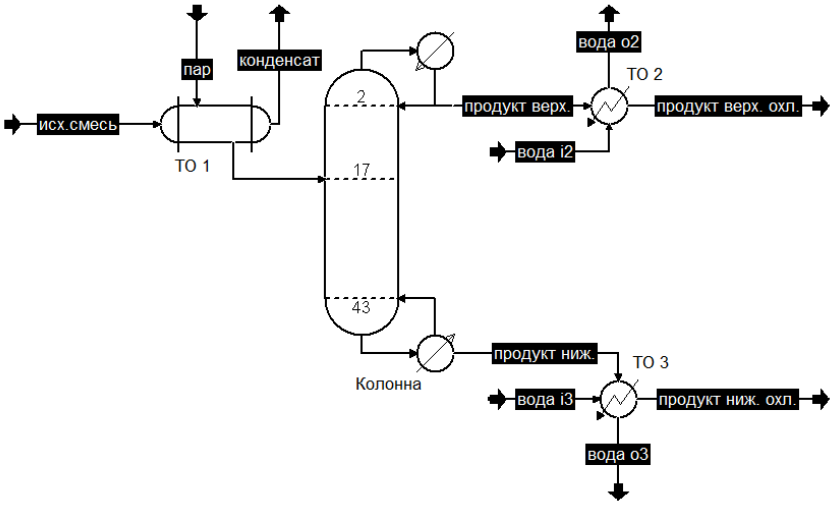


Рис. 3. Схема разделения бинарной смеси

В результате расчетов получены значения параметров верхнего и нижнего продуктов (рисунок 4). Расход дистиллята составил 6,64434 т/ч, с температурой 61,3845 °С и содержанием хлороформа 0,9548 мольных долей. Расход остатка – 7,35566 т/ч с температурой 77,5401 °С и содержанием хлороформа 0,05 мольных долей.

Stream	исх.смесь	исх.смесь подогретая	продукт верх.	продукт ниж.	Unit
Pressure	1	1	1	1	atm
Temperature	30	76	61.3845	77.5401	°C
Flow rate	14	14	6.64434	7.35566	ton / h
Mole frac chlorof	0.395	0.395	0.9548	0.05	
Mole frac benze	0.605	0.605	0.0452	0.95	

Рис. 4. Значения параметров верхнего и нижнего продуктов

В таблице параметров по потокам теплообменников, приведенной на рисунке 5, показаны температуры охлажденных продуктов и охлаждающей воды.

Stream	продукт верх.	вода i2	продукт верх. охл.	вода o2	продукт ниж.	вода i3	продукт ниж. охл.	вода o3	Unit
Pressure	1	1	1	1	1	1	1	1	atm
Temperature	61.3845	18	25	25.6462	77.5401	18	25	38.7871	°C
Flow rate	6.64434	8.1	6.64434	8.1	7.35566	8.1	7.35566	8.1	ton / h

Рис. 5. Параметры потоков теплообменников

Таблицы расчетов параметров колонны содержат температуры и давление на каждой тарелке, флегмовое число и коэффициент кипения (паровое число) (рисунок 6).

Stage	Temperatur (C)	Pressure (atm)	Flow rates (kmol/s) Liquid	Vapour
1	62.4300	1.00000	0.0679663	RR=4.89106
2	62.8500	1.00000	0.0678694	0.0818623
3	63.4100	1.00000	0.0677450	0.0817654
. . . . .				
41	75.0200	1.00000	5.73354	3.53418
42	75.5700	1.00000	5.66142	3.47760
43	76.2700	1.00000	5.57284	3.40548
44	77.1100	1.00000	BR=1.47030	RR=0.70212

Рис. 6. Коэффициент рефлюкса и коэффициент кипения

В результате оптимизации встроенной функцией FUG получены 24 тарелки, подача на 11, рефлюкс 5,3 (рисунок 7).

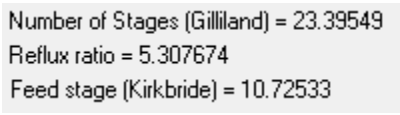


Рис. 7. Параметры программной оптимизации

Такая оптимизация работы ректификационной колонны приводит к сокращению теоретических ступеней разделения бинарной смеси хлороформ-бензол с одновременным увеличением расхода флегмы, подаваемой в колонну. Следовательно, повышая значение флегмового числа можно добиться разделения данной бинарной смеси на меньшем количестве контактных устройств, в аппарате, имеющем меньшую высоту. При этом также увеличивается расход по пару в верхней $G_в$и нижней части $G_н$ колонны, в зависимости от которого определяется диаметр колонны в верхней и нижней части $D_в$ и $D_н$.

$$G_{\text{В}} = \frac{P(R+1)\text{M}_{\text{ср.в.}}}{\text{M}_P};\ G_{\text{Н}} = \frac{P(R+1)\text{M}_{\text{ср.н.}}}{\text{M}_P};\ D_{\text{В}} = \sqrt{\frac{4G_{\text{В}}}{\pi\omega^{\text{В}}\rho_{\text{п}}^{\text{в}}}};\ D_{\text{Н}} = \sqrt{\frac{4G_{\text{Н}}}{\pi\omega^{\text{Н}}\rho_{\text{п}}^{\text{Н}}}}$$

На следующем этапе исследования работы ректификационной колонны провели сравнение расчетного метода определения оптимального флегмового числа, задавшись различными коэффициентами избытка флегмы и определив соответствующие флегмовые числа R. Для каждого значения R графическим построением ступеней изменения концентраций между равновесной и рабочими линиями в диаграмме состав пара y – состав жидкости x определяется количество теоретических тарелок $N_{т.}$ Результаты расчетов приведены в таблице 1.

Таблица 1 - Данные для построения графика

R	$N_{теор.}$	$N_{теор.}(R+1)$
1,812	31	87,17
2,265	28	91,42
2,718	24	89,23
3,171	21	87,6
3,624	20	92,48

По данным из таблицы 1 строится график зависимости N(R+1)=f(R) (рисунок 8), на котором минимальному значению N(R+1) соответствует оптимальное флегмовое число R.

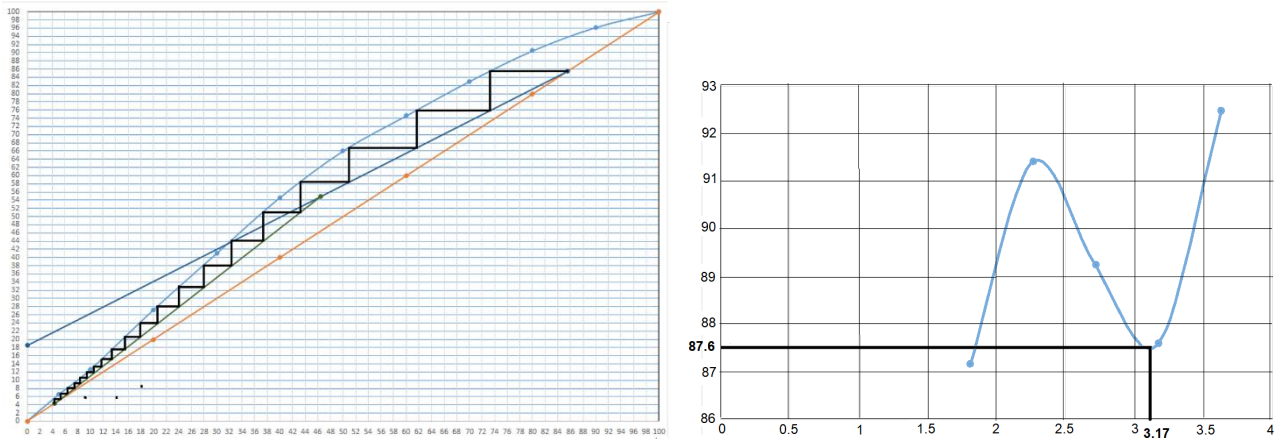


Рис. 8. Определение оптимального флегмового числа

Оптимальное флегмовое число R=3,171, теоретическое количество тарелок $N_{теор.}$ =21.

В модели колонны задается оптимальное количество тарелок. На рисунке 9 приведены параметры колонны и флегмовое число, полученное в результате работы модели. Флегмовое число модели на 0,061 меньше расчетного.

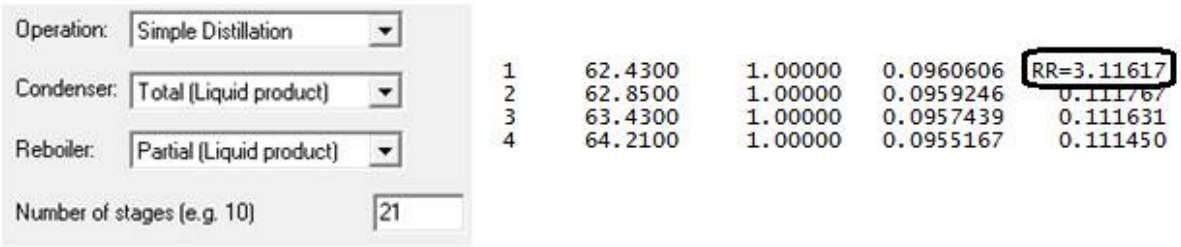


Рис. 9. Результат оптимизации

Таким образом, исследование работы ректификационной колонны в программе CAPE-OPEN to CAPE-OPEN simulation позволяет, задавшись расходом и составом питания и дистиллята, флегмовым числом и давлением в колонне, определить нагрузку колонны по пару и жидкости и ее диаметр, построением рабочих линий процесса ректификации найти оптимальное число ступеней разделения и высоту аппарата.

Библиографический список

1. Коваленко В.В., Кулавина Н.Ю., Шашкина Г.А. Филиппова А.В. Программные продукты для моделирования химико-технологических систем в учебном процессе / Современные технологии в науке и образовании – СТНО-2016: сб. тр. между-нар. науч.-техн. и науч.-метод. конф.: в 4 т. Т.3./ под общ. ред. О.В. Миловзорова. – Рязань: Рязан. гос. радиотехн. ун-т, 2016. – С. 155-158.

2. Коваленко В.В., Кулавина Н.Ю., Шашкина Г.А. Ветшев К.А., Рубцова А.Д. Моделирование разделения нефтяных фракций с использованием гипотетических компонентов / Современные технологии в науке и образовании –СТНО-2021: сб. тр. между-нар. науч.-техн. и науч.-метод. конф.: в 10 т. Т.3./ под общ. ред. О.В. Миловзорова. –Рязань: Рязан. гос. радиотехн. ун-т, 2021. – С. 61-65.

3. Руководство COCO [Электронный ресурс] – Режим доступа: https://www.cocosimulator.org/.

УДК 628.33 ; ГРНТИ 70.25.17

ИССЛЕДОВАНИЕ ОЧИСТКИ СТОЧНЫХ ВОД ОТ КОМПЛЕКСОВ ТЯЖЕЛЫХ МЕТАЛЛОВ

Е.Р. Корнешова*, А.В. Порхунова*, Е.В. Воробьева**
**Рязанский государственный радиотехнический университет имени В.Ф. Уткина,*
Российская Федерация, Рязань, katyakorneshova@mail.ru
***Рязанский институт (филиал) Московского политехнического университета*
Российская Федерация, Рязань, vorobeva_70@bk.ru

Аннотация. В работе рассматривается метод очистки сточных вод от комплексов тяжелых металлов. Приводятся особенности его использования и данные полученные при исследовании.
Ключевые слова: радиоэлектронная промышленность, сточные воды, экология, методы очистки, ионообменный метод, лабораторные испытания.

INVESTIGATION OF WASTEWATER TREATMENT FROM COMPLEXES HEAVY METALS

E.R. Korneshova*, A.V. Porkhunova*, E.V. Vorobyova**
**Ryazan State Radio Engineering University named after V.F. Utkin,*
Russia, Ryazan, katyakorneshova@mail.ru
***Ryazan Institute (branch) Moscow Polytechnic University*
Russian Federation, Ryazan, vorobeva_70@bk.ru

The summary. The paper considers the method of wastewater treatment from heavy metal complexes. The features of its use and the data obtained during the study are given.
Keywords: radioelectronic industry, waste water, ecology, purification methods, ion exchange method, laboratory tests.

Радиоэлектронная промышленность является одним из ключевых направлений современной промышленности, основой высокотехнологичных изделий многих отраслей производства. В любой конечной продукции присутствуют или электронные компоненты, или радиоэлектронные узлы. Все предприятия радиоэлектронного кластера используют при производстве своей продукции электрохимические методы, в частности гальванику, в результате чего образуется большое количество сточных вод, содержащих целый спектр разнообразных загрязняющих веществ, которые могут образовывать прочные комплексы.

Данная тема является актуальной, так как существует ряд процессов, когда в электролитах или технологических растворах присутствуют вещества-комплексообразователи, такие как Трилон Б, аммиак, аминоуксусная кислота, цианиды и т.д. Эти вещества специально вводят в раствор для максимально долгого сохранения целевых компонентов (металлов и их соединений) в растворенном состоянии. В этом случае, многие металлы очень легко образуют с этими веществами комплексные (анионные, катионные и нейтральные), а также хелатные соединения [1]. Экологическая опасность растворов, в которых содержатся такие комплексы тяжелых металлов очень высока. При неверно организованном водоотведении это приводит к тому, что большая часть сбрасываемых в коллектор высокотоксичных примесей приходится на долю отработанных электролитов и технологических растворов. Из-за этого возникает острая необходимость в поиске новых методов очистки сточных вод, в которых могут содержаться анионные и хелатные комплексы тяжелых металлов и при этом не будут превышены нормы ПДК.

В настоящее время существуют различные методы очистки сточных вод, которые подразделяются на биологические, химические, электрохимические и физико-химические. Каждая группа методов включает в себя несколько способов очистки, но обычными методами разрушить такие комплексы очень сложно, необходимы жесткие методы окисления или восстановления. Для этого рассматриваются такие методы, как озонирование, которое, однако, имеет ряд существенных недостатков: очень дорогостояще; озон является токсичным,

следовательно необходим тщательный контроль техники безопасности; непродолжительность воздействия. Окисление активным хлором также имеет недостатки, в первую очередь токсичность и канцерогенность. Можно восстанавливать эти металлы из комплексов электролизом, однако метод целесообразен при небольшом объеме сточных вод. При большом объеме сточных вод это крайне энерго- и экономически затратно [2].

Для извлечения металлов из подобных комплексных соединений свою эффективность показал ионный обмен на фильтрах, заполненные анионитом или катионитом. Каждая смола различается по свойствам, имеет свой фильтроцикл и еще ряд определенных требований. После достижения фильтроцикла смолу необходимо регенерировать для возврата её к первоначальным свойствам и дальнейшему применению по очистке воды. Регенерационные растворы – элюаты, содержат в себе все извлеченные из воды анионы или катионы в концентрированном виде. Эти элюаты также необходимо переработать для извлечения вредных компонентов и последующей утилизации.

По данному методу были проведены исследования по эффективности очистки сточных вод от комплексных соединений никеля и железа, которые они образуют с трилоном Б и аминоуксусной кислотой, с использованием ионообменных смол различных марок.

Для подбора смолы было необходимо установить следующие параметры:

– определить содержание металлов в сточной воде;

– определить оптимальный pH максимального осаждения металлов;

– определить способность каждого вида смолы сорбировать металлы, присутствующие в воде и связанные в комплексы с Трилоном Б и аминоуксусной кислотой;

– экспериментально определить фильтроцикл выбранной смолы.

Для определения оптимальных условий процесса осаждения было изучено влияние pH на полноту осаждения никеля и железа. При повышении pH от 9,5 до 11 наблюдалось значительное увеличение и укрупнение осадка, а также ускорение скорости оседания частиц, что является важными параметрами при выборе оптимальных условий проведения процесса осаждения. Наиболее полно никель осаждается при pH 11,0, то же самое характерно и для железа.

Для определения количества никеля и железа, связанных в комплексы, в пробы воды, поступающей на очистные сооружения, в стехиометрическом количестве были добавлены Трилон Б и аминоуксусная кислота. Затем последовательно проведено осаждение гидроксидов металлов при различных значениях pH. Количество осадка, полученного после осаждения, даже при pH 11,0, существенно меньше, чем в отсутствие комплексов. На скорость оседания хлопьев присутствие комплексообразователей не влияет. Это справедливо как для Трилона Б, так и для аминоуксусной кислоты. Данные представлены в таблице 1.

Таблица 1. Влияние pH на процесс осаждения гидроксидов металлов

Тип воды	Концентрация Me на входе в ЛОС, мг/дм^3		$pH_{вход}$	pH 9,5	pH 10,0	pH 10,5	pH 11,0	ПДК
Сточная вода (1)	$[Ni^{2+}]$	2,30	3,12	0,35	0,16	0,09	0,05	0,25
	$[Fe^{3+}]$	0,35		0,14	0,12	0,17	0,03	5
Сточная вода + Трилон Б (2)	$[Ni^{2+}]$	2,30	3,12	2,30	2,29	2,21	2,12	0,25
	$[Fe^{3+}]$	0,35		0,35	0,24	0,17	0,07	5
Сточная вода + аминоуксусная кислота (3)	$[Ni^{2+}]$	2,30	3,12	2,14	2,14	2,04	1,97	0,25
	$[Fe^{3+}]$	0,35		0,17	0,21	0,13	0,10	5

При добавлении в воду трилона Б осаждение никеля происходит плохо даже при pH 11, что свидетельствует об образовании очень прочных хелатных комплексов, на которые даже сильное защелачивание раствора не оказывает воздействия.

В случае аминоуксусной кислоты процесс осаждения протекает немного лучше, но закономерность сохраняется, и остаточное содержание никеля в воде также существенно превышает ПДК. С железом аминоуксусная кислота, по всей видимости, образует менее прочные комплексы, осаждение проходит достаточно полно во всем исследованном интервале pH.

Следующим этапом предстояло определить, какая смола наиболее эффективно показывает себя при очистке от комплексов никеля и железа в каждом конкретном случае. Для исследований были выбраны следующие образцы (Trilite MA-12, Trilite MC-10H, Lewatit Monoplus TP 207).

Так как в отсутствие комплексов процесс осаждения гидроксидов металлов проходит очень хорошо, и при pH от 10,0 до 11,0 наблюдалось остаточное содержание металлов существенно ниже уровня ПДК, было решено рассмотреть сорбцию пробы воды, полученной при осаждении при pH 9,5. Для изучения процесса сорбции воды, содержащей комплексы никеля и железа, были выбраны пробы воды после осаждения при pH 11,0 как для трилона Б, так и для аминоуксусной кислоты.

Ионообменную сорбцию проводили следующим образом. Через ионообменную колонку, заполненную определенным типом смолы (объём смолы = 150 $см^3$), последовательно был пропущен объем воды выбранных образцов, равный 1 $дм^3$, для качественного и количественного определения пригодности каждого типа смолы для очистки сточной воды в присутствии комплексов металлов и без них.Результаты сорбции приведены в таблице 2.

Таблица 2. Определение типа смолы, пригодной для очистки от комплексных соединений никеля и железа

Тип воды на входе в колонку	Содержание Me на входе в колонку, мг/$дм^3$		Trilite MC-10H	Trilite MA-12	Lewatit Monoplus TP 207	ПДК
Сточная вода (1)	$[Ni^{2+}]$	0,35	0,005	0,34	0,07	0,25
	$[Fe^{3+}]$	0,14	<0,001	0,12	<0,001	5
Сточная вода + Трилон Б (2)	$[Ni^{2+}]$	2,12	2,12	<0,001	2,12	0,25
	$[Fe^{3+}]$	0,07	0,05	<0,001	0,01	5
Сточная вода + аминоуксусная кислота (3)	$[Ni^{2+}]$	1,97	1,96	<0,001	0,009	0,25
	$[Fe^{3+}]$	0,10	0,09	0,05	<0,001	5

Из полученных данных при существовании никеля и железа в воде в виде катионов эффективность доочистки показывают катионообменные смолы. В присутствии комплексона трилона Б очистка воды возможна только с использованием анионообменной смолы. Смолы Trilite MA-12 и Lewatit Monoplus показали эффективную сорбционную очистку от комплексов никеля и железа с аминоуксусной кислотой.

Далее следовало определить фильтроцикл смолы Trilite MA-12 при сорбции воды, содержащей комплексы никеля и железа с Трилоном Б. Определение фильтроцикла проводилось на модельных растворах. В связи с тем, что смола Trilite MA-12 анионообменная, её ёмкость также будет заполняться существующими в воде анионами. Фильтроцикл составил 21 $дм^3$, суммарное содержание по никелю и железу, сорбированное смолой, 22,6 мг.

Аналогичным образом на модельных растворах был определен фильтроцикл смолы Lewatit Monoplus при сорбции воды, содержащей комплексные соединения никеля и железа с аминоуксусной кислотой. Фильтроцикл смолы по никелю составил 22 дм^3, для железа он будет значительно больше. Суммарное содержание по никелю и железу, сорбированное смолой, 56,8 мг.

В связи с высоким солесодержанием и ввиду специфики в сточных водах присутствует достаточное количество катионов щелочных металлов, соли и гидроксиды которые хорошо растворимы. При проведении процесса осаждения они остаются в растворенном состоянии и во время сорбции также занимают часть ёмкости смолы.

Очищенная вода, подаваемая на выход, не содержит загрязняющих веществ и может быть использована в качестве промывной воды для отмывки ионообменных колонн после проведения процесса регенерации.

При использовании катионообменных смол металлы в элюатах присутствуют в виде катионов, их последующее выделение в виде малорастворимых соединений не представляет сложности.

В случае с комплексными соединениями процесс обработки и последующей утилизации элюатов требует проведения дополнительных операций по предварительному разрушению этих комплексов. Возможна химическая обработка элюатов растворами гипохлорита натрия, перекиси водорода и других реагентов с последующим осаждением гидроксидов металлов. Однако, предположительно, наибольшей эффективности при обработке элюатов возможно достичь при применении электрохимического восстановления металлов. Комплексные соединения под действием электрического тока будут разрушаться, и на катоде будет происходить восстановление металлов.

Для определения оптимального способа переработки элюатов, полученных при регенерации смол Trilite MA-12 и Lewatit Monoplus, необходимо проведение дальнейших целенаправленных исследований.

Таким образом, в лабораторных условиях были произведены исследования по очистке модельных растворов сточных вод с аминоуксусной кислотой и Трилоном Б на различных ионообменных смолах. Было показано, что наиболее эффективная очистка от комплексов металлов с аминоуксусной кислотой достигается при использовании в процессе сорбции смолы Lewatit Monoplus TP 207, а для очистки комплексов металлов с Трилоном Б – смолы Trilite MA-12. Остаточное содержание тяжелых металлов в очищенных стоках составляет менее 0,1 мг/дм^3 по никелю и железу, что соответствует нормам ПДК.

Библиографический список

1. Наумов В.И., Мацулевич Ж.В., Ковалева О.Н. Комплексные соединения. – Нижний Новгород: НГТУ им. Р.Е. Алексеева, 2019.

2. Халтурина Т.И. Очистка сточных вод промышленных предприятий. – Красноярск: СФУ, 2014.

МЕЖДУНАРОДНАЯ НАУЧНО-ТЕХНИЧЕСКАЯ КОНФЕРЕНЦИЯ «СОВРЕМЕННЫЕ ТЕХНОЛОГИИ В НАУКЕ И ОБРАЗОВАНИИ. ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ»

СЕКЦИЯ «ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В КОНСТРУИРОВАНИИ ЭЛЕКТРОННЫХ СРЕДСТВ»

УДК 004.4; ГРНТИ 50.05

ИСПОЛЬЗОВАНИЕ СОВРЕМЕННЫХ СТЕКОВ ВЕБ-ТЕХНОЛОГИЙ ДЛЯ РАЗРАБОТКИ ИНФОРМАЦИОННЫХ СИСТЕМ В ПРОЕКТИРОВАНИИ И ПРОИЗВОДСТВЕ ЭЛЕКТРОННЫХ СРЕДСТВ

А.Н. Сапрыкин, А.А. Храмова, Т.С. Васильева
Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, saprykin.a.n@rsreu.ru

Аннотация. В работе рассматриваются современные стеки веб-технологий для разработки информационных систем, их роль в процессе проектирования и производства электронных средств.
Ключевые слова: информационная система, веб-технологии, SPA, DOM, препроцессор, фреймворк, язык программирования, архитектура ИС.

USE OF MODERN WEB TECHNOLOGY STACKS FOR THE DEVELOPMENT OF INFORMATION SYSTEMS IN THE DESIGN AND PRODUCTION OF ELECTRONIC TOOLS

A.N. Saprykin, A.A. Khramova, T.S. Vasileva
Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, saprykin.a.n@rsreu.ru

The summary. The paper considers modern stacks of web technologies for the development of information systems, their role in the design and production of electronic means.
Keywords: information system, web technologies, SPA, DOM, preprocessor, framework, programming language, IS architecture.

На сегодняшний день глобальная конкуренция такова, что производство любых электронных средств не может продолжаться, если продукция не обладает преимуществами перед аналогами: например, высокое качество или низкая цена. При таких условиях многие производители идут путем наименьшего сопротивления, используя большой объем памяти компьютера, высокую скорость обработки и удобные для пользователя возможности интерактивной графики для автоматизации и связывания воедино громоздких и отдельных инженерных или производственных задач, тем самым сокращая время и стоимость разработки и производства электронных средств. Компьютерное проектирование (CAD), автоматизированное производство (CAM) и автоматизированное проектирование (CAE) – это технологии, используемые для этой цели в ходе производственного цикла. Данные системы играют главную роль в процессе проектирования и производства электронных средств. Помимо данных систем в данном процессе используются такие системы как MES, ERP и OLAP. Современный уровень развития облачных технологий позволяет переносить процесс разработки и проектирования в облако, тем самым повышая актуальность использования веб-технологий.

Стек веб-технологий – это набор языков программирования, библиотек, фреймворков и программного обеспечения, используемых для разработки информационных систем (ИС). От выбранного разработчиком стека технологий зависит скорость, надежность работы, безопасность данных, устойчивость к высоким нагрузкам и масштабируемость ИС.

Любое веб-приложение имеет две основные составляющие. Одна доступна пользователю и отвечает за пользовательский интерфейс, в то время как другая отвечает за работу приложения. Видимая пользователю сторона, которая используется им для связи с приложением – фронтенд или пользовательские функции и интерфейс. Стек фронтенд-технологий в основном включает HTML, CSS, JavaScript, TypeScript или фреймворки на основе JS, такие как React, Vue и Express. Бэкенд составляющая приложения или его серверная часть отвечает за работу приложения в целом. Она недоступна пользователю, а технология, отвечающая за ее создание, известна как стек бэкенд-технологий. В основном они представлены языками программирования, библиотеками, структурами, серверами, системами администрирования наборов данных и т. д.

Разработка клиентской части

Для разработки клиентской части используются следующие технологии [3]:

1. HTML – стандартизированный язык разметки, помогающий структурировать содержание веб-приложения и правильно отображать его в браузере;

2. CSS – язык таблицы стилей, помогающий настроить визуальное оформление веб-приложения, а также расположить объекты на экране в определенном порядке и плоскости.

Для повышения функциональности языка CSS создано множество вариантов надстроек для модификации.

CSS-препроцессор – это программа, которая генерирует CSS-код, используя собственный синтаксис, который уникален для каждого препроцессора. Данные программы преобразуют код, написанный на препроцессорном языке, в чистый и валидный CSS-код. Помимо этого, они используются для добавления CSS-функций, которые изначально отсутствуют в языке. Далее рассмотрим наиболее популярные CSS-препроцессоры.

SASS. Предоставляет расширенные возможности CSS, используется в программирования для стилевого оформления страниц. Это один из наиболее известных CSS-препроцессоров. SASS обладает множеством функций и совместим со всеми версиями языка CSS. Использует расширение «.sass».

Переменные в препроцессоре SASS используются для хранения любых свойств CSS: размера шрифтов, цвета, семейства шрифтов и т. д. Стоит отметить, что в данном препроцессоре названию переменной всегда предшествует символ «$».

Примеси в SASS задействуются в первую очередь ради возможности многократного использования стилей для селекторов.

Также важной функциональной возможностью SASS является импорт. В SASS возможно получить доступ к коду из другого файла с расширением «.sass» и импортировать не только данные, находящиеся в файле того же каталога, но и содержимое файла из другого каталога.

LESS. Позволяет использовать переменные, функции, циклы и другие технологии для упрощения работы со стилями. Главное его преимущество перед остальными препроцессорами – это простота, близкая схожесть со стандартным синтаксисом и набор плагинов для расширения функциональности. Использует расширение «.less».

Одной из главных особенностей данного препроцессора является возможность создания переменных, как в стандартном языке программирования. Они могут хранить любые типы данных: размеры, цвета, имена шрифтов, URL-адреса и т. д. LESS позволяет применять стили уже существующего класса или идентификатора к другому селектору.

Stylus. Данный препроцессор прост и удобен в использовании. Конструкция синтаксиса схожа с CSS, но в коде можно не прописывать фигурные скобки, точки с запятой после каждого свойства, а также двоеточия после названия свойства. Использует расширение «.styl». Stylus позволяет создавать переменные, но при этом не использует символ для их

обозначения. Назначение переменной выполняется автоматически путем разделения свойства и ключевых слов.

«Примеси» Stylus работают аналогично «примесям» Sass и LESS. Их можно использовать для хранения и повторного использования наборов правил пользовательского стиля. Однако прозрачные «примеси» являются уникальной особенностью Stylus. Они используются как свойства в CSS.

Stylus имеет встроенные функции для обработки и преобразования цветов и единиц измерения, вычисления средних, минимальных и максимальных значений.

CSS-фреймворки – это наборы классов и свойств, которые облегчают процесс стилизации веб-приложения. Рассмотрим наиболее популярные CSS-фреймворки.

Bootstrap – фреймворк, который активно используется веб-разработчиками для повышения скорости вёрстки сайтов и веб-приложений. Он обладает следующими преимуществами:

- адаптивная сетка, основанная на Flex-модели, которая решает большую часть задач компоновки содержимого веб-приложения;
- множество шаблонов и готовых компонентов. Для большинства интерактивных компонентов написан готовый JavaScript код;
- поддержка препроцессоров SASS и LESS.

Foundation – это CSS фреймворк, построенный на HTML, CSS, SASS и JavaScript с mobile-first подходом, с полностью адаптивными компонентами. В качестве основных преимуществ следует отметить: адаптивная сетка (в создании дизайна по функциональности не уступает Bootstrap), простая стилизация и полный контроль интерфейса проекта, поддержка JavaScript-компонентов, легкое создание анимаций, вертикальный контроль разметки, поддержка препроцессора SASS.

Tachyons – CSS фреймворк, основной целью которого является разделение правил CSS на несколько небольших, управляемых частей. Впоследствии эти части можно использовать повторно.

Преимущества Tachyons:

- элементы фреймворка сразу готовы к применению. Он фокусируется на обеспечении классов утилит, позволяющие повысить производительность работы приложения. В документации описано множество готовых к работе функций;
- есть функциональные шаблоны, которые подходят для любых условий, например, к статическому HTML, React, Angular и др.;
- многогранное применение;
- хорошо подходит для прототипирования;
- необязательно соблюдать стили по умолчанию.

JavaScript – язык программирования, позволяющий создавать в веб-приложении интерактивные элементы. Для данного языка программирования существует множество фреймворков и библиотек, которые позволяют создавать различные части пользовательских интерфейсов, чтобы повторно использовать эти компоненты независимо друг от друга, создавая высокоинтерактивные веб-приложения. Рассмотрим несколько современных JavaScript фреймворков.

React – это популярная библиотека JavaScript для создания масштабируемых веб-приложений, особенно в тех ситуациях, когда приложение представляет одностраничное приложение (SPA). В React возможно использование двух синтаксисов для создания компонентов: классовый и функциональный. Классовый синтаксис почти не используется, т.к. нередко оказывается избыточен для описания компонентов.

Функциональный компонент – JavaScript-функция с входными данными для отрисовки компонента и выходными, такими как описание интерфейса, который нужно отобразить. В результате не нужно описывать, как перерисовывать интерфейс SPA. Вместо этого необ-

ходимо указать, что нужно отобразить на месте компонента, и использовать для этого синтаксис, схожий с HTML. Такой подход позволяет относительно легко решать и более сложные задачи. Например, перерисовать разметку страницы корзины и отобразить на её месте интерфейс страницы заказа.

React использует виртуальную объектную модель документа (DOM). Это позволяет обновлять только изменившиеся узлы DOM-дерева. Благодаря этому приложение затрачивает меньше времени и ресурсов на перезагрузку страницы.

Angular – это фреймворк от компании Google, предназначенный, в первую очередь, для разработки SPA. Главное отличие фреймворка Angular от других – использование языка программирования TypeScript. Angular включает в себя:

- набор встроенных библиотек, которые предоставляют следующие функции: маршрутизацию, взаимодействие клиент-сервер, управление формами и т. д.;
- множество компонентов для разработки масштабируемых веб-приложений;
- набор инструментов, с помощью которых становится проще создавать и тестировать код веб-приложения.

Помимо уже перечисленного, одним из основных преимуществ использования данного фреймворка является двусторонняя привязка. Такая структура позволяет мгновенно синхронизировать модель и представление. Так разработчикам проще вносить изменения сразу во время разработки.

Angular.js построен на архитектуре MVC (Model – View – Controller), которая широко используется для разработки полнофункциональных веб-приложений.

VueJS – это прогрессивный JavaScript фреймворк с открытым исходным кодом, предназначенный для разработки пользовательского интерфейса.

Рассмотрим функции VueJS:

- Использование виртуальной DOM. Как и упоминалось ранее в описании функций библиотеки React, виртуальная DOM позволяет экономить ресурсы и время на обновление компонентов страницы с помощью сохранения изменений в виртуальной DOM и последующим сравнением исходных данных.
- Директивы. С помощью различных встроенных директив можно выполнять ряд действий во Frontend. Возможно присваивание и управление значениями HTML-атрибутов, присваивание классов с помощью директивы v-bind.
- Прослушивание событий. С помощью атрибута v-on, который добавляется к компонентам DOM, можно прослушивать события, происходящие в веб-приложении.
- Для создания переходов и анимации можно воспользоваться встроенным компонентом VueJS. Можно также легко добавить сторонние библиотеки анимации.
- Маршрутизация. С помощью атрибута vue-router можно настроить навигацию между частями веб-приложения.
- Компактность кода. Скрипт фреймворка очень компактный, что обеспечивает высокую производительность.

Разработка серверной части

Для разработки серверной части используются следующие технологии: различные принципы разработки (YAGNI, KISS, DRY и др.); языки бэкенд-программирования (PHP, Go (для хайлоад проектов) с применением шаблонов проектирования); фреймворки: Symfony, Laravel, Zend Framework для создания и поддержки развития новых проектов; Yii и Kohana для совершенствования готовых сторонних или собственных проектов; базы данных (в качестве реляционной выступает – MySQL, а документарной – MongoDB); сервер. Возможна реализация веб-приложения с использованием бессерверной архитектуры [2].

Рассмотрим подробнее основные принципы разработки чистого кода для создания ИС.

Принцип YAGNI (You are not gonna need it) – в переводе с англ. «Тебе это не понадобится» – подразумевает, что при разработке кода реализуются только нужные для функционала участки кода, игнорируя написание важных в теории фрагментов. Таким образом, программисту не требуется осуществлять добавление нового функционала, а лишь править ненужный код.

Принцип KISS (Always Keep It Simple, Stupid) – в переводе с англ. «Всегда делай это проще» – подразумевает, что при написании кода будут использоваться небольшие методы около 40-50 строк, каждый из которых будет решать одну проблему. Таким образом, система работает наилучшим образом, поскольку не усложнена без необходимости, например, установкой целой библиотеки ради одной ее функции. В силу того, что код написан надежно, при изменении его фрагментов в будущей перспективе не возникнет сложностей.

Сказанное выше можно реализовать в React для сокращения кода при использовании стрелочных функций, деструктуризации, async/await синтаксиса, использовании блоков try/catch для обеспечения надежности.

Принцип DRY (Don’t Repeat Yourself) – в переводе с англ. «Не повторяйся» – подразумевает, что при создании кода программист будет добавлять различные константы, выносить общую логику и избегать копирования фрагментов кода. Также важным аспектом является проверка написанного функционала в проекте перед добавлением нового.

Данный принцип можно выразить в React при использовании в коде переиспользуемых компонентов, на которых и базируется принцип React. В свою очередь, пользовательский интерфейс нужно создавать из переиспользуемых блоков.

Рассмотрим языки бэкенд-программирования, использующиеся при разработке ИС.

PHP – это серверный скриптовый язык (1997 год), который используется в сфере веб-разработки и обеспечивает работу примерно 78,9% всех веб-сайтов [1].

Код PHP чаще всего интерпретируется, обрабатывается и отображается с помощью веб-сервера с установленным модулем PHP, что позволяет встраивать в HTML-разметку код, который находится в файлах с расширением .php.

К особенностям языка PHP можно отнести:

- простоту и общеизвестность – свойства, сделавшие его одним из ведущих языков программирования по изучению и пользованию;
- платформонезависимость – язык, работающий на всех операционных системах (ОС) и платформах;
- свободную типизацию – тип данных переменной в PHP не обязательно прописывать, поскольку он будет указан из расчета данных, которые хранятся в ней во время выполнения программы;
- интерпретируемость языка – компиляция перед выполнением программного кода не обязательна, поскольку осуществляется построчное выполнение строки одна за другой и вывод результата;
- гибкость – возможна легкая встройка и интеграция языка с такими скриптовыми языками, как XML, JavaScript и HTML для создания приложения;
- множество веб-фреймворков – преимущество их использования в том, что программист может писать структурированный код в структуре MVC, что помогает определить функциональные возможности в конкретной структуре. К примеру, фреймворки: CodeIgnitor, Laravel и др.

Ограничения языка PHP:

1. За счет открытого доступа к исходному коду языка нельзя злоупотребить слабостями скрипта.

2. Плохой вариант для крупных проектов в связи со слабым обслуживанием.

3. Отсутствие доступа к специальным библиотекам и некорректная обработка ошибок.

Go – ценный статически типизированный язык бэкенда. Он был выпущен в 2009 году. Создатели старались разработать эффективный при компиляции и выполнении программ язык программирования, который позволит с легкостью писать программы повышенной сложности. ОН имеет некоторое сходство с языком программирования С, предназначен для достижения максимума с минимальными затратами. В нем сочетаются хорошие идеи из других языков без усложнения кода. Исходные тексты библиотек и инструментов, а также компиляторы языка размещены в открытом доступе, так что каждый программист может внести свой вклад в разработку его библиотек и инструментов. Однако, несовершенная обработка ошибок в языке Go способна вызвать проблемы как у программистов, так и у пользователей приложений.

Рассмотрим подробнее некоторые PHP-фреймворки, которые используются при разработке серверной части ИС.

Symfony – отличается от других фреймворков надёжностью и опытом (2005 год). Он стремительно обрел популярность за счет соответствия веб-стандартам, паттернам проектирования PHP. Он имеет встроенную систему тестирования и основан на крупном MVC-фреймворке. Отметим, что подобные составляющие делают его хорошим вариантом для крупных веб-проектов на уровне организаций. Symfony, в частности, может работать с разными БД.

CodeIgniter – фреймворк, который прост в освоении, удобен в применении, имеет подробную документацию и активное сообщество, а для приведения его в рабочее состояние требуется минимум настроек (2006 год). Однако он медленно развивается в плане использования современных технологий и шаблонов и по мере прогресса фреймворков и совершенствования инструментов PHP-разработки начал отставать и с точки зрения технологий, и с точки зрения предлагаемых по умолчанию функциональных возможностей. В отличие от многих других фреймворков CodeIgniter разрабатывался компанией, поэтому поддержка новых возможностей PHP 5.3, таких как пространства имен или использование GitHub, а позднее – Composer, внедрялась медленно. Он является хорошим варинтом для реализации динамических веб-сайтов в силу своей компактности (размер примерно 2 Мб).

Zend Framework – это фреймворк, который был придуман и впоследствии реализован в 2006 году как надежная и функциональная библиотека компонентов для разработчиков на PHP, представляет собой одно из ответвлений от проекта CodeIgniter. Он позволяет быстро и эффективно решать множество распространенных задач, связанных с разработкой приложений, таких как создание и проверка форм ввода, обработка XML, генерация динамических меню, разбивка данных на страницы, работа с веб-сервисами и другое.

Yii – это производительный универсальный фреймворк, имеющий в своей основе компонентную структуру (2008 год). Он известен своей простотой. Yii позволяет использовать код сторонних разработчиков, а встроенный в него генератор кода Gii позволяет стремительно создавать базовые структуры для построения собственных решений. В данный фреймворк встроены механизмы хеширования паролей, базирующиеся на bcrypt, и средства шифрования.

Laravel – это фреймворк для быстрой разработки, являлся одним из альтернатив проекту CodeIgniter. Он предельно прост в освоении и сводит к минимуму количество шагов между запуском нового приложения и его публикацией. Его компоненты упрощают разработку веб-приложения: от взаимодействия с БД и аутентификации до работы с очередями, электронной почтой и кэшем. Laravel предоставляет целую экосистему инструментов для создания и запуска приложений. Например, Homestead и Valet для локальной разработки, Forge для управления серверами и Envoyer для расширенного развертывания, а также набор дополнительных пакетов: Cashier для платежей и подписок, Echo для веб-сокетов, Scout для поиска, Passport для API – аутентификации и т.п.

Рассмотрим некоторые подходы для построения серверной архитектуры ИС.

1. Архитектура микросервисов.

Микросервисы представляют собой шаблон, организующий приложение как набор мелкомодульных сервисов слабой связи, которые осуществляют взаимодействие по упрощенным протоколам. Данная архитектура является современным подходом к написанию программных приложений (ПП) из набора сервисов. Каждый сервис создается с помощью особого набора технологий и ориентирован на конкретную задачу или бизнес – функцию. Подобный подход предоставляет гибкость для инженерных групп и бизнеса.

Многие компании пользуются микросервисами для своих продуктов и услуг, например, Amazon, Sound Cloud, eBay, Netflix, Microsoft, Uber и т. д.

2. Бессерверная архитектура.

Бессерверная архитектура является хорошим выбором, если при создании приложения или сервиса нет необходимости в управлении инфраструктурой. Удобство состоит в том, что администрирование сервера, на котором работает приложение, переходит под руководство облачного провайдера (GCP, AZURE, AWS и др.). Будет сохранено большое количество времени и ресурсов, поскольку перекладывается ответственность за масштабирование и обслуживание используемого сервера. Пользователи могут получить доступ к бизнес-логике приложения через сервер. Забота об обслуживании серверного оборудования, программного обеспечения и обновлениях безопасности закрепляется за командой. Наконец, разработчики будут полностью сосредоточены на написании кода приложения.

Модель BaaS делает возможным для разработчика сфокусироваться на задачах разработки интерфейса. К примеру, AWS Amplify – известный продукт BaaS. Нельзя не отметить, модель FaaS, которая управляется событиями. За счет нее разработчики разбивают приложение на функции, чтобы сфокусироваться на триггерах событий и коде. Остальное будут обрабатывать поставщики услуг FaaS, К примеру, AWS Lambda и Microsoft Azure – поставщики услуг FaaS, которые будут осуществлять обработку прочих задач.

Многие компании используют гибридную стратегию при разработке бессерверных технологий, задействуя и другие технологии.

Таким образом, следует отметить, что рассмотренные стеки веб-технологий, представляющие собой совокупность языков программирования, библиотек, фреймворков и программного обеспечения, используемых для разработки ИС, таких как CAD, CAM, CAE, MES, ERP и OLAP, в силу современных тенденций играют все большую и большую роль в процессе проектирования и производства электронных средств.

Библиографический список

1. Кисленко, Н. П. Интернет-программирование на PHP: учебное пособие / Н. П. Кисленко. – Новосибирск: Новосибирский государственный архитектурно-строительный университет, ЭБС АСВ, 2015. – 177 с.

2. Кудряшев, А. В. Введение в современные веб-технологии: учебное пособие / А. В. Кудряшев, П. А. Светашков. – 3-е изд. – Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. – 359 с.

3. Сычев, А. В. Перспективные технологии и языки веб-разработки: практикум / А. В. Сычев. – 3-е изд. – Москва: Интернет – Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. – 493 с.

УДК 004.4; ГРНТИ 50.05

ИСПОЛЬЗОВАНИЕ СОВРЕМЕННЫХ РЕЛЯЦИОННЫХ СУБД ДЛЯ РАЗРАБОТКИ ОБЛАЧНЫХ САПР ЭЛЕКТРОННЫХ СРЕДСТВ

А.Н. Сапрыкин, Т.С. Васильева, А.А. Храмова
Рязанский государственный радиотехнический университет имени В.Ф. Уткина,
Российская Федерация, Рязань, saprykin.a.n@rsreu.ru

Аннотация. В работе рассматриваются современные реляционные СУБД для разработки облачных систем автоматизированного проектирования электронных средств. Анализируются их основные преимущества и недостатки, приводятся аргументы в пользу хранения данных в облаке.
Ключевые слова: САПР, СУБД, база данных, облачные технологии.

USE OF MODERN RELATIONAL DBMS FOR THE DEVELOPMENT OF CLOUD CAD OF ELECTRONIC TOOLS

A.N. Saprykin, T.S. Vasileva, A.A. Khramova
Ryazan State Radio Engineering University named after V.F. Utkin,
Russia, Ryazan, saprykin.a.n@rsreu.ru

The summary. The paper considers modern relational DBMS for the development of cloud systems for computer-aided design of electronic means. Their main advantages and disadvantages are analyzed, arguments are given in favor of storing data in the cloud.
Keywords: CAD, DBMS, database, cloud technologies.

Понимание того, как использовать и применять программное обеспечение САПР, является фундаментальным навыком для всех студентов, изучающих инженерные науки и дизайн. Важно, чтобы студенты изучали не только фундаментальные основы, но и новые и появляющиеся практики, принятые в отрасли. Это включает в себя, например, растущую интеграцию с технологиями PLM/PDM или встроенные инструменты CAE. Инженеры и дизайнеры редко работают в одиночку, и часто необходимо работать вместе с партнерами и соавторами над сборкой CAD для работы над проектом. Нередко возникают проблемы с коллаборацией, когда разработчики получают доступ удаленно (локально или глобально), используют разные файлы САПР и не работают в общем цифровом пространстве. Возможным решением данной проблемы становится новая тенденция в САПР – облачные интерфейсы, где большая часть (если не все) трехмерного моделирования деталей и сборок выполняется через веб-браузер. В качестве преимуществ таких систем стоит отметить удобство пользования и редактирования деталей и сборок, а также облачную совместную работу с другими пользователями – в том числе, облачное хранение данных.

Для хранения и управления данными больших объемов в организованном виде в современных информационных системах используются базы данных (БД). Управление ими осуществляется системами управления базами данных (СУБД), которые представляют собой тип программного обеспечения, позволяющий пользователю создавать, поддерживать и редактировать базы данных. Например, осуществление удаления и добавления данных, нахождение и смена структуры необходимых элементов. СУБД поддерживает языки баз данных и отвечает за копирование и восстановление информации после сбоев. Хранение БД в облаке повышает безопасность и надежность процесса проектирования электронных средств, защищая разработчика от случайной утраты данных, а также дает возможность дистанционной работы сразу нескольких пользователей над одним проектом.

Реляционные БД

Реляционные системы управления базами данных (РСУБД) – это системы, базирующиеся на теории множеств. Отметим, что основой таких систем служат двумерные таблицы. Каноническим способом взаимодействия с РСУБД является написание запросов на языке

Structured Query Language (SQL). Значения данных типизированы (например, строки, числа, даты, неструктурированные двоичные объекты и т.п.). Контроль типа данных осуществляет система. Благодаря теории множеств исходные таблицы можно соединять и модифицировать в более объемные [3].

Существует немало реляционных СУБД с открытым исходным кодом. Рассмотрим некоторые из них.

MySQL – это производительная реляционная СУБД с большим количеством API (1995 год) [1]. Обращение к серверу СУБД могут осуществлять разные клиентские приложения, а значит, она имеет клиент-серверную архитектуру. Обладание весомыми функциональными возможностями позволяет решать разноплановые задачи.

Преимущества MySQL:

- кроссплатформенность (Mac OS, Windows, Linux, Solaris и др.);
- открытый исходный код;
- наличие большого количества API, благодаря чему возможно подключение к базе проектов на таких языках, как Java, Python, PHP и др.;
- достойные конструктивные возможности (масштабируемость, быстродействие, многопоточность, многопользовательский доступ);
- наличие развитой системы разграничения доступа, а также системы обеспечения безопасности.

В качестве ограничений MySQL следует отметить возникновение сложностей при работе с большими базами, несоответствие стандарту SQL, низкая скорость развития приложения.

SQLite – это библиотека программного обеспечения, осуществляющая бессерверный и автономный механизм базы данных SQL, основанный на транзакциях (2000 год). За счет наличия нулевой конфигурации в механизме, она не требует отдельной настройки в системе. На данный момент это один из самых быстро развивающихся движков баз данных в мире. Движок не автономен, а значит можно осуществить его статическое, либо динамическое связывание в зависимости от потребностей системы. SQLite имеет прямой доступ к своим файлам хранения. Стандартные команды SQLite для взаимодействия с реляционными базами подобны языку SQL.

Преимущества SQLite:

- для работы не требует отдельный серверный процесс (бессерверный);
- не требует администрирования или настройки (нулевая конфигурация);
- база данных хранится в одном кроссплатформенном файле на диске;
- мал и легковесен, полная конфигурация составляет менее 400 Кб;
- полная совместимость транзакций с ACID, тем самым предоставляется безопасный доступ из нескольких потоков;
- легкость и простота в использовании API;
- доступность в UNIX-подобных системах и Windows;
- исходный код для SQLite находится в открытом доступе.

Ограничениями SQLite можно считать отсутствие правого и полного внешнего соединений (реализовано только левое внешнее соединение), отсутствие возможности внесения изменений в таблицы (удаление и изменение столбца, добавление ограничения – поддерживаются только следующие варианты команды ALTER TABLE – переименование таблицы и добавление столбца), поддержка триггеров для каждой строки, но не для каждого оператора, доступность представления только для чтения, запрет использовать над ними инструкции удаления, вставки или обновления, применение лишь обычных разрешений на доступ к файлам базовой ОС.

PostgreSQL – многофункциональная реляционная СУБД с открытым кодом (1989 год) [2]. Она применима к бизнес-системам, которые работают под большой нагрузкой. Вклад разработчиков из России в проект весьма значителен.

Преимущества PostgreSQL:

- надежность и устойчивость в работе с важными данными;
- безопасность (например, подключение по SSL-соединению, аутентификация по паролю и с помощью внешних сервисов, использование клиентских сертификатов и др.);
- полное соответствие таким стандартам, как ANSI SQL, SQL92-SQL:2016;
- поддержка свойств ACID и обеспечение отделения транзакций с помощью механизма MVCC;
- разработчик получает в собственное распоряжение богатый инструментарий для реализации приложения любого вида;
- масштабируемость и производительность;
- возможности индексирования (Hash, GiST, GIN, SP-GiST, RUM, Bloom, BRIN);
- кроссплатформенность и доступность;
- расширяемость (например, возможность добавлять типы данных, функции, подключения к внешним источникам данных, загружаемые расширения и др.);
- независимость (не принадлежит компаниям, развивается программистами различных сообществ).

Ограничения PostgreSQL: подходит не для всех задач, в которых особо важна скорость выполнения операций; наличие только транзакционных таблиц; трудности при инсталляции.

MariaDB является достаточно известной СУБД, которая была разработана разработчиками MySQL. Отметим, что поддержка в обработке данных, как для малых, так и для больших частных задач подкупает. Она также считается достойной заменой MySQL, которая требует лишь его деинсталляции и установки MariaDB. Данная СУБД отличается от MySQL набором новых уникальных функций.

Преимущества MariaDB:

- нахождение всех приложений под лицензиями LGPL, GPL, либо BSD;
- обладание широким спектром механизмов хранения, требующихся для работы с иными источниками данных СУБД;
- использование сегментированного кеша ключей;
- возможность параллельного выполнения нескольких запросов;
- поддержка разных ОС и языков программирования;
- поддержка языка PHP и кластерной технологии Galera;
- наличие многих команд, операций, ныне отсутствующих в MySQL, а также устранение/замена функций, понижающих производительность.

MariaDB обладает и недостатками, а именно: неуверенность в стабильности работы СУБД в связи с ее открытым исходным кодом; большие траты для поддержки ПО.

Графические клиенты

Графический клиент – это инструмент, позволяющий пользователю запрашивать и визуализировать данные, хранящиеся в БД, а также управлять ими и анализировать. С помощью графического клиента СУБД можно получать доступ к множеству серверов БД и перемещаться по ним.

Существует множество графических клиентов для СУБД. Рассмотрим некоторые из них.

PgAdmin – популярное графическое средство для администрирования PostgreSQL. Программа упрощает основные задачи администрирования, отображает объекты БД, позволяет выполнять запросы SQL.

PgAdmin обладает рядом особенностей, а именно: совместимостью со многими ОС: Linux, Windows, macOS, возможностью одновременной работы с несколькими серверами, функцией экспорта файлов в формате CSV, возможностью отслеживания сессий, блокировки БД, наличием встроенного компилятора процедурного языка и хорошо написанной документацией.

PhpMyAdmin – графический клиент, написанный на языке PHP. Он поддерживает множество операций с такими СУБД, как MySQL и MariaDB. С помощью PhpMyAdmin можно производить операции по управлению структурой БД, напрямую выполнять любую инструкцию SQL. С помощью удобного графического интерфейса PhpMyAdmin можно легко просмотреть структуру БД, различных таблиц и полей.

В качестве особенностей phpMyAdmin стоит отметить возможность выполнения SQL-запросов, совместимость с множеством ОС, простоту использования, возможность экспортировать и импортировать данные в различных форматах, администрирование БД с помощью графического приложения и открытый исходный код.

DBeaver – инструмент, предназначенный для управления БД, такими как PostgreSQL, MySQL, MariaDB и другие JDBC-совместимые БД. С помощью удобного графического интерфейса DBeaver возможно просматривать структуру БД, исполнять SQL-запросы, экспортировать данные и т.д. Графический клиент – интегрированный, для него необходима поддержка языка Java.

Особенности DBeaver:

- поддержка любых БД, имеющих драйвер JDBC;
- управление различными сторонними источниками данных;
- подсветка синтаксиса, автозаполнение SQL;
- расширенный набор плагинов и утилит для управления БД;
- визуализация индивидуальных объектов базы и ее схемы, изменение схемы;
- экспорт данных в различные форматы, такие как CSV, HTML, XML, XLS, XLSX;
- возможность генерации пробных данных для тестирования БД;
- редактирование данных в онлайн формате.

Таким образом, использование облачных САПР электронных средств позволяет разработчикам снижать затраты на оборудование и повышать надежность хранения данных за счет использования облачных БД. Они являются удобными решениями для совместных проектов и семинаров. С точки зрения эффективности облачные САПР в целом могут обеспечить более гибкий и универсальный доступ к программному обеспечению САПР в учреждении. Реляционные СУБД поддерживают данную технологию, давая возможность разработчикам электронных средств хранить данные в облаке, быстро и удобно получать к ним доступ с разных компьютеров и в целом делать процесс проектирования электронных средств эффективнее. Дальнейшее улучшение существующих облачных САПР электронных средств во многом может быть осуществлено за счет грамотного использования существующих СУБД.

Библиографический список

1. Введение в СУБД MySQL: учебное пособие. – 3-е изд. – Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. – 228 с.

2. Гутман, Г. Н. Объектно-реляционная СУБД PostgreSQL: учебное пособие / Г. Н. Гутман. – Самара: Самарский государственный технический университет, ЭБС АСВ, 2016. – 125 с.

3. Иванова, О. Г. Управление данными. Использование технологий ORACLE для реализации баз данных: учебное пособие / О. Г. Иванова, Ю. В. Кулаков, С. В. Данилкин. – Тамбов: Тамбовский государственный технический университет, ЭБС АСВ, 2021. – 81 с.

УДК 004.896; ГРНТИ 47.14

РЕАЛИЗАЦИЯ КОНФИГУРИРУЕМЫХ ПРОЦЕССОРОВ НА ОСНОВЕ КОНЕЧНЫХ АВТОМАТОВ

М.С. Кошелева

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, igoshina.m.s@rsreu.ru

Аннотация. В работе рассматривается возможная реализация конфигурируемого процессора в базисе ПЛИС. Описывается принцип работы устройства и его структура, основанная на использовании конечных автоматов.

Ключевые слова: конфигурируемы процессоры, ПЛИС, конечные автоматы, HDL- проектирование.

IMPLEMENTATION OF CONFIGURABLE PROCESSORS BASED ON FINITE AUTOMATA

M.S. Kosheleva

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, igoshina.m.s@rsreu.ru

The summary. The paper considers the possible implementation of a configurable processor in the FPGA basis. The principle of operation of the device and its structure based on the use of finite automata are described.

Keywords: configurable processors, FPGAs, state machines, HDL- design.

Считается, что ПЛИС представляет собой вспомогательный инструмент к микропроцессору, способный эффективно реализовывать дополнительные операции, а также обеспечивать определённый запас гибкости для построения цифровых интерфейсов.

Однако в последнее время ПЛИС, благодаря модификациям технических характеристик, позволяющих им увеличивать функциональность процессоров, плотно закрепились в статусе «система на кристалле».

Система на кристалле или однокристальная система представляет собой электронную схему, реализующую функционал полноценного устройства, размещенная на одной интегральной схеме. В зависимости от назначения она может оперировать как цифровыми сигналами, так и аналоговыми, аналого-цифровыми, а также частотами радиодиапазона. Довольно часто применяются в портативных и встраиваемых системах. За счёт своих функциональных ресурсов ПЛИС позволяют получить макет цифрового устройства, например soft–процессора, в кратчайшие сроки с возможностью неограниченного внесения изменений без затрат материалов или комплектующих.

В данной статье будут рассмотрен один из возможных способов реализации конфигурируемого процессора в базисе ПЛИС, основанный на применении двух конечных автоматов.

ПЛИС

В настоящее время ПЛИС стали ископльзовать в качестве самостоятельной среды проектирования сложных цифровых систем. Подобная тенденция объясняется возможностью многократного переконфигурирования архитектуры на кристалле, путем программирования соединений между универсальными логическими блоками (ячейками). На одном кристалле могут быть реализованы как аппаратные преобразователи, решающие конкретные технические задачи, так и программная реализация, имеющая процессорное ядро и набор периферии. Данная возможность позволяет разрабатывать оптимальные алгоритмы для специфических задач, связанных с параллельными и повторяющимися вычислениями. Возможность переконфигурирования собственной архитектуры в отличие от микропроцессоров и заказных интегральных схем (СБИС), делает ПЛИС универсальным инструментом.

Ещё одним достоинством ПЛИС является возможность параллельного выполнения десятков и сотен операций, что увеличивает производительность ПЛИС по отношению к микропроцессорам в разы. В настоящее время стоимость ПЛИС может быть в десятки раз больше чем стоимость микропроцессоров, а производительность нередко оказывается значительно ниже, чем у специализированных СБИС, однако многофункциональность ПЛИС сделала её конкурентоспособной. ПЛИС, в качестве элементной базы, удачно применяется в случаях, когда скорость микропроцессора является недостаточной, а процесс разработки заказной СБИС чересчур дорогостоящий и длительный [3].

Все перечисленные особенности ПЛИС делают ее универсальным инструментом для обработки больших потоков данных в режиме реального времени, в том числе и видеоинформации.

Для реализации конфигурируемого процессора остановим свой выбор на ПЛИС Cyclone IV EP4CE6E22C8N от фирмы Altera, представленной на рисунке 1, и обладающей следующими характеристиками:

- рабочая частота: 50 MHz,
- рабочее напряжение: 1.15 V?3.465 V,
- корпус: QFP144,
- количество портов I/O: 80,
- LE: 6K,
- RAM: 270 kb,
- PLL: 2,
- программирование/отладка**:** поддержка интерфейса JTAG и т.д.

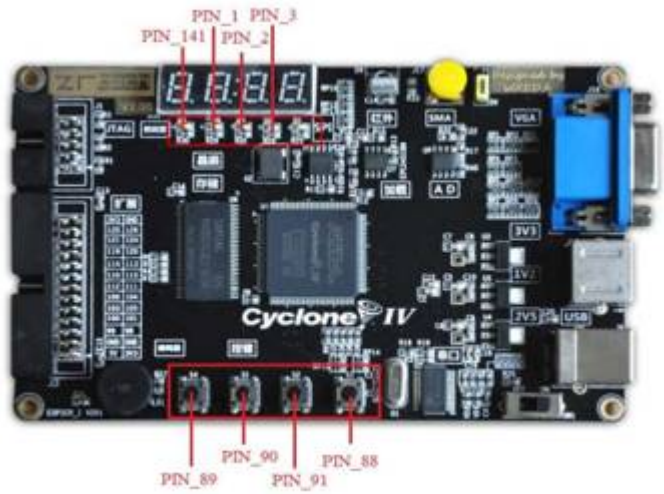


Рис. 1. ПЛИС Altera Cyclone IV

Устройства Cyclone IV обладает диапазоном плотностей, способных обеспечивать различное количество логических ресурсов устройства, включая LE, память и множители. Определение необходимой логической плотности представляет собой неотъемлемую часть процесса планирования проекта. Устройства с большим количеством логических ресурсов могут реализовывать более крупные и потенциально более сложные конструкции, но обычно имеют более высокую стоимость. Меньшие устройства имеют более низкое статическое потребление энергии. Устройства Cyclone IV поддерживают вертикальное перемещение, что обеспечивает гибкость.

При использовании ПЛИС Cyclone IV, в качестве элементной базы, процесс проектирования необходимо начать с создания подробной спецификации проекта для системы и ПЛИС

и определения интерфейса ввода и вывода для остальной части системы. Перед созданием проекта или завершением проектирования системы подробные спецификации проектирования должны определить систему, указать интерфейс ввода-вывода для ПЛИС, идентифицировать различные тактовые домены и включить блок-схему основных функций проектирования. Затраты времени для создания этих спецификаций позволяют повысить эффективность проектирования.

Далее необходимо создать план функциональной проверки, чтобы можно было проверить работу системы. Создание плана тестирования на этом этапе также помогает повысить эффективность и технологичность. При этом может понадобиться возможность проверить все интерфейсы проекта.

Программирование осуществлялось на высокоуровневом языке VHDL, обладающим следующими особенностями:

- Проекты цифровых устройств имеют иерархическую структуру;
- Каждый автономный проектируемый модуль имеет определенный интерфейс взаимодействия с другими модулями и точную спецификацию внутреннего устройства проектируемого модуля, который описывает концепцию и функционирование модуля;
- Спецификация модулей VHDL-проектов могут использовать математические алгоритмы, которые описывают их работу или же описание аппаратной структуры проектируемого модуля [2];
- Моделирование алгоритма работы проекта основывается на событийном принципе управления;
- Позволяет выполнять моделирование процессов в электрических схемах, временной анализ сигналов и их параметров;
- VHDL является языком параллельного программирования, то есть в его конструкции есть операторы, соответствующие логическим вентилям.

В качестве программного обеспечения для выбранной ПЛИС, будем использовать САПР Intel Quartus Prime от фирмы Altera. Подобная мультисреда предназначена для создания устройств с высокой степенью интеграции, совместима с выбранной отладочной платой. Благодаря своему богатому функционалу, система Quartus Prime обеспечивает возможность реализации всех этапов разработки устройства (например, проектирования и синтез, размещение элементов, трассировку соединений и верификацию) в одном пакете, а также поддерживает связь с системами проектирования других производителей.

Конфигурируемый процессор на основе конечных автоматов

Разрабатываемый конфигурируемый процессор в базисе ПЛИС представляет собой калькулятор, реализующий простейшие арифметические операции над шестнадцатеричными числами, такие как: сложение, вычитание, деление и умножение.

Принцип действия калькулятора основан на работе двух конечных автоматов (приема и передачи). Конечные автоматы — цифровые устройства, в которых обратные связи включены, таким образом, где в каждый момент времени выходное состояние такого устройства зависит не только от входных воздействий, но и от состояний выходов в предыдущий момент времени [1].

1. Конечный автомат приема отвечает за прием данных (значений операндов – каждый операнд по 8 бит). Каждое поступившее значение сохраняется в отдельный регистр, после чего передается код арифметической операции и аналогично сохраняется в собственный регистр. Исходя из поступившего кода операции, где «0» – сложение, «1» – вычитание, «2» – умножение, «3» – деление, производится соответствующее вычисление. Результат вычисле-

ний сохраняется в регистре результата. После чего формируется сигнал для конечного автомата передачи. Работу автомата приема отображает сигнал *rx*.

2. Конечный автомат передачи возвращает полученный результат вычислений в виде двух восьмибитных чисел на экран монитора. Передача сигнала осуществляется в 2 шага – по одному восьмибитному числу за раз. Работу автомата приема отображает сигнал *tx*.

Значения операндов (2 чисел по 8 бит) и код арифметической операции вводятся с клавиатуры. Передача данных, а также результата вычисления между ПЛИС и ЭВМ осуществляется при помощи UART-интерфейса.

Взаимодействие с подобным интерфейсом реализуется за счет программы *Terminal* – простейшего монитора COM-порта персонального компьютера, представленного на рисунке 3, позволяющего свободно передавать необходимые данные через COM-порт ЭВМ по протоколу RS-232. В качестве достоинств пакета можно выделить:

- возможность перенастройки программы в зависимости от выбранного режима работы;
- наличие элементарного интерфейса, подходящего для любого пользователя;
- наличие счетчика принятых и переданных байтов;
- возможность поддерживания до 64 COM-портов и т.д.

VHDL-код разрабатываемого устройства:

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use ieee.numeric_std.all;
entity main is
    generic(
    N : integer := 2604 -- 50_000_000 / 9600 / 2
    );
    Port ( clk : in  STD_LOGIC;
           rx : in  STD_LOGIC;
           tx : out  STD_LOGIC;
           LD0 : out  STD_LOGIC;
           reset : in  STD_LOGIC);
end main;

architecture Behavioral of main is
type  t_state_rx  is  (idle_rx,  start_rx,  data_rx,  stop_rx,
arithm, nope_rx);
type  t_state_tx  is  (idle_tx,  start_tx,  data_tx,  stop_tx,
count_res_tx);
signal state_rx, state_next_rx : t_state_rx := idle_rx;
signal state_tx : t_state_tx := idle_tx;
signal clk_tx : std_logic := '0';
signal data_0 : signed(7 downto 0);
signal data_1 : signed(7 downto 0);
signal operation : unsigned(7 downto 0);
signal res : signed(15 downto 0);
signal strttx : std_logic := '0';

begin
```

```
p_rx: process(reset, rx, clk)
    variable count_rx : integer := 0;
    variable data_i : integer := 0;
    variable clk_i : integer := 0;
    variable n_i : integer := 0;
    variable m : integer := 0;
begin
    if reset = '1' then
        count_rx := 0;
        data_i := 0;
        clk_i := 0;
        n_i := 0;
        strttx <= '0';
        LD0 <= '0';

    elsif rising_edge(clk) then

        case(state_rx) is

            when idle_rx =>
                if rx = '0' then
                    state_rx <= start_rx;
                end if;

                strttx <= '0';
                data_i := 0;
                clk_i := 0;

            when start_rx =>

                m := 3;
                state_rx <= nope_rx;
                state_next_rx <= data_rx;

            when data_rx =>
                m := 2;

                if count_rx = 0 then
                    data_0(data_i) <= rx;
                elsif count_rx = 1 then
                    data_1(data_i) <= rx;
                elsif count_rx = 2 then
                    operation(data_i) <= rx;
                end if;

                if data_i = 7 then
                    state_next_rx <= stop_rx;
                else
                    data_i := data_i + 1;
                end if;
```

```
                    state_rx <= nope_rx;

                when stop_rx =>

                    if count_rx = 2 then
                        state_next_rx <= arithm;

                    else
                        count_rx := count_rx + 1;
                        state_next_rx <= idle_rx;
                    end if;

                    m := 3;
                    state_rx <= nope_rx;
                when arithm =>

                    if operation = x"00" then
                        res  <=  to_signed(0,  16)  +  data_0  +
data_1;
                    elsif operation = x"01" then
                        LD0 <= '1';
                        res  <=  to_signed(0,  16)  +  data_0  -
data_1;
                    elsif operation = x"02" then
                        res  <=  to_signed(0,  16)  +  data_0  *
data_1;
                    elsif operation = x"03" then
                        res  <=  to_signed(0,  16)  +  data_0  /
data_1;
                    end if;

                    m := 2;
                    strttx <= '1';
                    count_rx := 0;
                    state_rx <= nope_rx;
                    state_next_rx <= idle_rx;

                when nope_rx =>

                    if clk_i = N - 1 then

                        if n_i = m - 1 then
                            n_i := 0;
                            state_rx <= state_next_rx;
                        else
                            n_i := n_i + 1;
                        end if;

                        clk_i :=  0;
                    else
                        clk_i := clk_i + 1;
```

```
                    end if;
               end case;
          end if;

     end process;

     p_tx: process(reset, clk_tx)
     variable data_i : integer := 0;
     variable res_i : integer := 0;
     begin

          if reset = '1' then
               res_i := 0;
               data_i := 0;
               --LD0 <= '0';
               tx <= '1';
          elsif rising_edge(clk_tx) then

               case(state_tx) is

                    when idle_tx =>

                         if strttx = '1' then

                              state_tx <= start_tx;
                         end if;

                         data_i := 0;
                         tx <= '1';

                    when start_tx =>
                         tx <= '0';
                         state_tx <= data_tx;

                    when data_tx =>

                         if res_i = 0 then
                              tx <= res(data_i);
                         else
                              tx <= res(data_i + 8);
                         end if;

                         if data_i = 7 then
                              data_i := 0;
                              state_tx <= stop_tx;
                         else
                              data_i := data_i + 1;
                         end if;

                    --   LD0 <= '1';
```

```
                when stop_tx =>
                     tx <= '1';
                     state_tx <= count_res_tx;
                when count_res_tx =>
                     if res_i = 1 then
                          res_i := 0;
                          state_tx <= idle_tx;
                     else
                          res_i := res_i + 1;
                          state_tx <= start_tx;
                     end if;
          end case;

     end if;
end process;

p_clk_tx: process(clk)
variable i : integer := 0;
begin

     if rising_edge(clk) then
          if i = N then
               i := 0;
               clk_tx <= not clk_tx;
          else
               i := i + 1;
          end if;
     end if;

end process;
end Behavioral;
```

Вывод

В данной статье был представлен один из возможных способов реализации конфигурируемого процессора на основе ПЛИС. Подробно рассмотрены главные особенности устройства, принципы проектирования, базирующиеся на применении высокоуровнего языка VHDL, а также технические возможности используемой отладочной платы. Кроме того предложено VHDL-описание процессора, представляющего собой калькулятор, выполняющий простейшие арифметические операции, такие как: сложение, вычитание, умножение и деление. Подобный принцип работы осуществляется за счёт использования двух конечных автоматов.

Библиографический список

1. Каширская Е. Н., Клягин М. М, Серебрянкин В. А. Теория конечных автоматов: учебное пособие / Е. Н. Каширская, М. М. Клягин, В. А. Серебрянкин. — Москва: РТУ МИРЭА, 2021. — 100 с.

2. Поляков А. К. Языки VHDL и VERILOG в проектировании цифровой аппаратуры / А. К. Поляков. — Москва: СОЛОН-Пресс, 2009. — 320 с.

3. Строгонов А. В. Реализация алгоритмов цифровой обработки сигналов в базисе программируемых логических интегральных схем: учебное пособие / А. В. Строгонов.— Санкт-Петербург: Лань, 2022. — 352 с.

УДК 004.21; ГРНТИ 27.41.41

ОПЕРАЦИЯ КРОССИНГОВЕРА В ГЕНЕТИЧЕСКОМ АЛГОРИТМЕ ЗАДАЧИ КОМПОНОВКИ КОНСТРУКТИВНЫХ МОДУЛЕЙ ЭЛЕКТРОННЫХ СРЕДСТВ

В.П. Корячко, А.О. Сапрыкина

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, saprykin.a.n@rsreu.ru

Аннотация. В данной работе показаны возможные варианты оператора кроссинговера генетического алгоритма, позволяющего решить задачу компоновки блоков большой размерности. Предложенные варианты процедуры скрещивания можно использовать как в классических генетических алгоритма, так и в гибридных.

Ключевые слова: генетический алгоритм, задача компоновки, конструкторское проектирование, кроссинговер.

CROSSOVER OPERATION IN THE GENETIC ALGORITHM FOR THE TASK OF DESIGNING OF STRUCTURAL MODULES OF ELECTRONIC MEANS

V.P. Koryachko, A.O. Saprykina

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, saprykin.a.n@rsreu.ru

The summary. The paper considers possible variants of the crossover operator of a genetic algorithm that allows solving the problem of arranging blocks of large dimensions. The proposed variants of the crossing procedure can be used both in classical genetic algorithms and in hybrid ones.

Keywords: genetic algorithm, layout problem, engineering design, crossover.

Этап конструкторского проектирования электронных средств (ЭС) определяется комплексом задач, связанных с преобразованием функциональных логических или принципиальных электрических схем в совокупность конструктивных узлов, между которыми устанавливаются необходимые пространственные, механические и электрические связи. При выполнении данного этапа также должны быть решены проблемы ограничения паразитных электромагнитных связей и учтены условия теплообмена. Конструкторский этап является завершающим в общем цикле разработки электронной аппаратуры и обеспечивает получение необходимой конструкторско-технологической документации для ее изготовления и последующей эксплуатации [3].

Одним из наиболее важных этапов проектирования ЭС является этап компоновки, который заключается в разделении функциональных элементов коммутационной схемы по группам, соответствующих конструктивным единицам разного ранга, из которых далее и проектируется ЭС. Под элементом, как правило, подразумевается конструктивная единица самого низкого ранга, а под блоком – самого высокого [1].

В [2] авторами показана зависимость числа вариантов компоновки от количества элементов в коммутационной схеме и формируемых блоков характеризует описанную задачу, как NP-полную. Для решения поставленной задачи было предложено использовать эволюционный генетический эволюционный алгоритм (ГА) оптимизации.

В настоящее время эволюционные принципы используются для решения прикладных задач оптимизации. Алгоритмы и методы, использующие в своей работе эволюционный подход, называют эволюционными. Самым распространенным классом эволюционных алгоритмов являются генетические алгоритмы.

ГА используются для решения широкого круга задач: численная оптимизация, поиск кратчайшего пути, компоновка, аппроксимация функций, нелинейная фильтрация, отбор данных и т.д.

ГА представляет собой стохастический эвристический алгоритм, применяющийся при решении задач оптимизация и выполнении моделирования при помощи последовательно подбора и рекомбинации, в основе которого лежат процессы, схожие с биологической эволюцией. В нем используется механизм генетического наследования и аналог естественного отбора. При этом сохраняется в упрощенном виде биологическая терминология и основные понятия линейной алгебры. Относится к области мягких вычислений.

Одним из наиболее важных этапов работы ГА является операция скрещивания или кроссинговер, который представляет собой рекомбинацию генов и обмен генетическим материалом родителей для получения потомков. Эта операция необходима для исследования новых и улучшения существующих областей пространства. Скрещивание считается главное операцией генетического алгоритма. Существует достаточно много вариантов выполнения данной процедуры.

Кроссинговер генетического алгоритма в задаче компоновки конструктивных модулей электронных средств

Одноточечное скрещивание

Пусть имеются две родительские хромосомы $p_1 = \{p_{1i}, i \in [0; N]\}$ и $p_2 = \{p_{2i}, i \in [0; N]\}$. Далее случайным образом выбирается целое число в заданном интервале. Полученное число называется точкой разрыва. В точке разрыва обе родительские хромосомы делятся на части и обмениваются ими, образуя тем самым двух потомков. Данный вид скрещивания получил название одноточечный, так как разрыв родительских хромосом происходит только в одной случайной точке.

Существует модификация одноточечного скрещивания, носящая название одноточечное скрещивание с возможностью копирования. Отличие данного подхода от рассмотренного состоит в том, что точка разрыва может принимать краевые значения интервала точек родителей, тем самым приводя к полному их копированию.

Двухточечное скрещивание

При данном варианте скрещивания хромосомы представляются в виде циклов, формируемых соединением концов линейной хромосомы вместе. Для того, чтобы заменить сегмент одного цикла сегментом другого необходимо сгенерировать две точки разрыва. Если рассматривать данную модель представления хромосом применительно к одноточечному кроссинговеру, то его можно представить как двухточечное скрещивание с одним фиксированный разрывом в начале строки. Из этого следует, что двухточечный кроссинговер решает задачу смешивания хромосом более полно, чем одноточечный, что также признается учеными и исследователями.

Многоточечное скрещивание

При многоточечном скрещивании выбирается m точек разрыва $k_i \in \{1,2,\ldots,N\}$, $i = 1:m$, N – число генов в хромосоме. Точки генерируются случайным образом, не содержат повторов и сортируются в порядке возрастания. При кроссинговере происходит последовательный обмен участками родителей, ограниченными точками разрыва, в результате чего формируется два потомка. Для воспроизведения при данном виде кроссинговера выбираются особи с наилучшей приспособленностью.

Однородное скрещивание

Для проведения данного типа скрещивания создается маска (схема скрещивания) индивидуумов. В каждом локусе маски содержится потенциальная точка кроссинговера. Длина схемы скрещивания совпадает с длиной родительских особей. Обычно при создании маски

кроссинговера используют следующий алгоритм. Вводится некоторая величина $0 < p_0 < 1$. Затем генерируют последовательность из N чисел от 0 до 1. Биты маски формируются по правилу, если i-ое сгенерированное число больше p_0, то i-ый элемент маски равен 0, иначе – 1. Потомки формируются на основе значений битов маски: если бит равен 0, то в потомка копируется бит текущей позиции первого родителя, иначе – второго. Для каждого создаваемого потомка генерируется своя схема скрещивания.

Однородный кроссинговер схож с многочечным, но строка случайных битов в нем гораздо длиннее, что гарантирует чередование коротких последовательностей генов родителей в потомке. Данный вид кроссинговера также называют унифицированным.

Триадный кроссинговер

Идея, положенная в основу триадного кроссинговера, схожа с однородным скрещиванием. Отличие состоит в том, что после выбора пары родителей из пула особей дополнительно отбирается третий индивидуум, который используется в качестве маски. После этого 10% генов особи-маски подвергаются мутации. Затем производится сравнение генов первого родителя с маской и, в случае, если они совпадают на сравниваемых позициях, передаются первому потомку. Второй потомок получает гены второго родителя, в ином случае они переходят на аналогичные позиции первого потомка.

Перестановочный кроссинговер

Согласно данному алгоритму особи-родители случайным образом обмениваются генами. При этом старые родители из пула удаляются, а их места занимают полученные в ходе обмена генами особи. Затем производят одноточечный кроссинговер и вновь «перетасовывают» гены потомков. Такой подход позволяет сократить число операций по сравнению с однородным кроссинговером

Кроссинговер с уменьшением замены

Уменьшение замены накладывает ограничения на операцию скрещивания, добиваясь создания новых особей всегда, когда это возможно. Это осуществимо за счет ограничения на выбор точки разрыва: она может появляться только там, где гены различаются.

Кроссинговер формирует особей-потомков на основе двух, уже имеющихся в популяции, комбинируя их гены между собой. Вследствие чего, многообразие различных комбинаций, создаваемых за счет скрещивания при использовании одной и той же пары готовых решений, ограничено и пространство, покрываемое генетическим алгоритмом только при использовании этой процедуры формирования потомков, жестко связано с генофондом популяции. От разнообразия генотипов решений популяции напрямую зависит размер пространства покрытия. Если находится локальный оптимум, соответствующий ему генотип будет стремиться занять как можно больше позиций в популяции. В таком случае возникает опасность схождения алгоритма к ложному оптимуму. Чтобы избежать этого сразу после применения операции кроссинговера используют операцию мутации.

BLX скрещивание

Обмен информацией производится с учетом значений генов родителей. Пусть имеются две родительские хромосомы $p_1 = \{p_{1i}, i \in [0; N]\}$ и $p_2 = \{p_{2i}, i \in [0; N]\}$. Значение i-го гена потомка выбирается случайным образом из интервала $[c_{\min} - \alpha\Delta_k, c_{\max} + \alpha\Delta_k]$, где α – константа, а c_{min} и c_{max} вычисляются по формулам:

$$c_{\min} = \min\{p_{1k}, p_{2k}\},$$
$$c_{\max} = \max\{p_{1k}, p_{2k}\},$$
$$\Delta_k = c_{\max} - c_{\min}.$$

SBX кроссовер

SBX (Simulated Binary Crossover) разработан в 1995 году. Он моделирует принципы двухточечного скрещивания. Идея данного метода строится на свойстве двухточечного кроссинговера о неизменности среднего значения функции приспособленности у родителей и их потомков, полученных в результате скрещивания. Было введено понятие силы поиска, представляющее собой количественную характеристику распределения вероятностей появления любого потомка от двух произвольных родителей. В результате работы данного оператора создаются два потомка в соответствии с формулой:

$$H_k = \left(h_{1k}, \ldots, h_{jk}, \ldots, h_{nk}\right),$$

где k=1,2,… – номер родителя;

j=1,2,…N – номер гена;

N – длина хромосомы;

h_{jk} – хромосомы потомки, создаваемые по заданным формулам с учетом родительских генов и коэффициентов на основе случайных чисел, распределенных по равномерному закону, а так же специального параметра кроссинговера – n.

Увеличение параметра n повышает шансы получения потомка в окрестности родителя.

При использовании оператора скрещивания важно не только корректно выбрать нужную его разновидность, но и задать подходящую вероятность применения. Стоит отметить, что результаты работы ГА напрямую зависят от выставленных параметров, наиболее важным из которых является правильный выбор кроссинговера. Параметры алгоритма в первую очередь влияют на исследование пространства поиска и то, как будут использованы найденные решения. То, как исследуется пространство поиска, отражает степень эффективности поиска решения и характеризует алгоритм с точки зрения способности избегать локальные оптимумы. Способ применения найденных решений отвечает за постепенное их улучшение в последующих поколениях. Суть корректной настройки алгоритма состоит в поиске баланса между указанными особенностями, в обратном случае может произойти увеличение времени работы алгоритма или ухудшение результатов, например, вследствие преждевременной сходимости, а игнорирование найденных приемлемых решений может превратить работу генетического алгоритма в случайный поиск.

Таким образом, генетические алгоритмы являются эффективным средством решения сложной задачи компоновки на этапе проектирования ЭС при условии грамотного и обоснованного выбора варианта оператора кроссинговера. Рассмотренные варианты скрещивания, а именно, одноточечное, двухточечное, многоточечное и однородное скрещивание, триадный кроссинговер, перестановочный кроссинговер, кроссинговер с уменьшением замены, SBX и BLX кроссинговер, позволяют сделать вывод о том, какие именно варианты наиболее эффективны для решения определенных задач. Описанные методы создавались в разное время с разными целями, в некоторых случаях привнося в принцип работы оператора скрещивания что-то новое, в других – заимствуя и улучшая предшественников. Основанное на известных преимуществах и недостатках различных методов знание принципов выбора нужного варианта оператора кроссинговера позволит повысить скорость и качество решения задачи компоновки конструктивных модулей электронных средств.

Библиографический список

1. Сапрыкин, А.Н. Алгоритмические методы автоматизации конструирования электронных средств: учебное пособие / А.Н. Сапрыкин. – Рязань: ИП Коняхин А.В. (Book Jet), 2021. – 116 с.
2. Сапрыкина, А.О., Сапрыкин, А.Н. Использование генетического алгоритма эволюционной оптимизации в задаче компоновки конструктивных модулей электронных средств // Материалы VII Всероссийской научно-технической конференции. – РГРТУ, Рязань, 2022. – С. 270-275.
3. Селютин, В.А. Машинное конструирование электронных устройств / В.А. Селютин. – Москва: Советское радио, 1977. – 384 с.

УДК 535.92; ГРНТИ 49.13

ПОВЫШЕНИЕ НАДЕЖНОСТИ ПЕРЕДАЧИ СООБЩЕНИЙ ПО ОПТИЧЕСКИМ БЕСПРОВОДНЫМ КАНАЛАМ

В.В. Яшкова*, М.С. Кошелева**

*Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, *yashkova.v.v@mail.ru, **igoshina.m.s@rsreu.ru*

Аннотация. В данной статье, на примере системы беспроводной передачи данных для коллектива мобильных роботов, рассматривается один из возможных способов повышения надежности передачи сообщений по оптическим беспроводным каналам за счет использования маркеров с улучшенным распознаванием элементов. Описываются недостатки и преимущества предложенного решения.
Ключевые слова: передача сообщений, оптические беспроводные системы передачи, оптические маркеры.

IMPROVING THE RELIABILITY OF MESSAGE TRANSMISSION OVER OPTICAL WIRELESS CHANNELS

V.V. Yashkova, M.S. Kosheleva

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, yashkova.v.v@mail.ru, igoshina.m.s@rsreu.ru

The summary. In this article, using the example of a wireless data transmission system for a team of mobile robots, one of the possible ways to increase the reliability of message transmission over optical wireless channels through the use of markers with improved element recognition is considered. The disadvantages and advantages of the proposed solution are described.
Keywords: message transmission, optical wireless transmission systems, optical markers.

В настоящее время популярность систем оптической беспроводной передачи данных значительно возросла. Это обусловлено тем, что они просты в реализации, а также обеспечивают высокий уровень надежности [1].

Область применения оптических беспроводных систем передачи очень разнообразна. Как правило, они используются для управления роботами или передачи данных между беспилотными летательными аппаратами (БЛА). Частным их случаем являются системы передачи видимым светом, например, с использованием светодиодов видимого излучения.

В данной статье описывается общий вид оптического маркера, способного упростить восприятие передаваемой информации, и тем самым позволяющего увеличить расстояние передачи данных.

Задача управления коллективом мобильных роботов

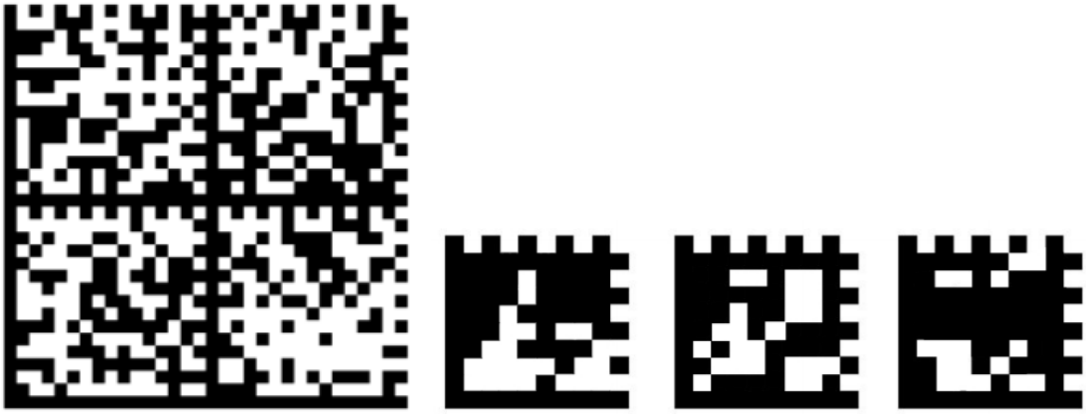
В современном мире задача управления коллективами роботов является актуальной, так как существует проблема установления информационного беспроводного общения роботов в группе. Решение подобного вопроса – построение системы оптического обмена инфор-

мацией с использованием технического зрения (СТЗ), где ключевой элемент – оптический маркер, позволяющий строить обмен информации между роботами [2].

Для кодирования и передачи данных о состоянии/положении робота применяют технологию двумерных кодов: оптические маркеры отображают сообщения в виде двумерных кодов, которые затем принимают видеокамеры СТЗ. Основное преимущество такого решения – элементарное распознавание кодов, что позволяет использовать их в самых различных сферах: от управления и контроля до бытового применения.

При разработке подобных систем следует учитывать потребность в увеличении информационной ёмкости двумерного кода, так как некоторые его фрагменты, предназначенные для надежного распознавания передаваемого сообщения, могут иметь большие размеры.

Данный недостаток возможно устранить за счет представления сообщений в нескольких последовательно отображаемых и передаваемых кадр за кадром двумерных кодах, составляющих вместе исходный объемный двумерный код (как показано на рисунке 1). Это позволит существенно (в 2,5 раза) увеличить дальность передачи сообщений при заданной вероятности их правильного декодирования по сравнению с передачей статическим двумерным кодом.



а) б)

Рис. 1. Пример представления оптического маркера в работе:
а) – исходный вид двумерного кода DataMatrix,
б) – динамический двумерный код исходного кода.

Оптический маркер

Замена статического двумерного кода на динамический с большим геометрическим размером некоторых фрагментов обеспечит возможность считывания с большего расстояния. Оптимальным решением проблемы увеличения дальности и повышения надежности рассматриваемых систем беспроводной связи является устранение принципиальных недостатков самого двумерного кода, в виде которого отображается передаваемое сообщение.

В качестве основного недостатка применяемого кода можно выделить применение решетчатой структуры из черно-белых штрихов. Подобный способ базируется на различении нескольких градаций толщины однородных по цвету штрихов, что усложняет процесс считывания и снижает его надежность.

Ещё одним значительным недостатком 2D-кодов является их координатная ориентированность, заключающаяся в том, что байты составляются из выстроенных в линии штрихов. Такая особенность нуждается в дополнительных преобразований при считывании кода, снижая скорость сканирования, соответственно.

Существуют алгоритмы распознавания двухмерных штриховых кодов, повернутых относительно камеры, однако их сложность ощутимо влияет на быстроту, необходимую для использования на мобильных системах.

Для увеличения уровня различимости и надежности сканирования, как правило, производят замену структуры с чередующимися темными и светлыми штрихами, а также разной шириной однородных штрихов на структуры с четкой привязкой геометрического места штрихов к определенным ориентирам.

В случае оптического маркера системой подобных ориентиров выступают расположенные равномерно по окружности светящиеся элементы, со светящимися опорными элементами (рис. 2). Из рисунка видно, что опорные светящиеся элементы расположены в вершинах наружного шестиугольника, синхронизирующие – в центре и по вершинам внутреннего шестиугольника, также подразумевается один дополнительный элемент (сверху справа), который является ключевым и задает направление начала обхода при считывании.

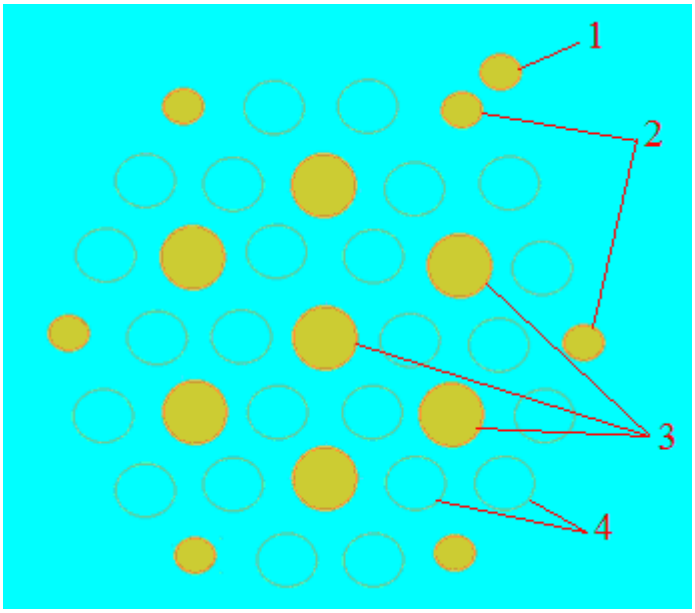


Рис. 2. Общий вид рассматриваемого оптического маркера:
1 – ключевой оптический элемент;
2 – опорные светящиеся элементы;
3 – синхронизирующие светящиеся элементы;
4 – световые элементы, кодирующие цифровую информацию.

Алгоритм передачи информационных данных с помощью оптического маркера

- Возникновение системы опорных точек, наведение устройства с камерой на оптический маркер. Задание уровня пороговой яркости, захват изображения со всеми граничными точками.
- Демонстрация на экране маркера изображения, сообщающего о готовности вывода информации. Такое изображение создается при 7 светящихся синхронизирующих элементах, которые добавляются к уже светящимся опорным элементам, то есть считывание системы светящихся элементов маркера происходит с последующего кадра и запоминание координат середин проекций светящихся областей на матрицу датчиков (номеров соответствующих ячеек).
- Вывод на экран маркера первого кадра информации. На этой итерации в ячейках памяти процессора видеокамеры фиксируются по определенным адресам биты, соот-

ветствующие освещенности (не освещенности) ячеек матрицы. Другими словами в матрице памяти устройства обработки фиксируется изображение, воспринимаемое от оптического маркера фотоприемниками.

- Идентификация и определение координат опорных элементов. Данный шаг является предшественником декодирования сообщения. Для его реализации следует в процессе сканирования матрицы фотодатчиков отличить опорные элементы от информационных и выделить их в отдельный кластер. Выделение опорных элементов базируется на их меньшем по сравнению с остальными элементами размере (для надежного различения диаметр опорных элементов должен быть примерно в 3 раза меньше). При сканировании нужно найти в строках матрицы более короткие группы (последовательности) засвеченных ячеек, после чего номер строки и столбца середины меньшей группы занести в таблицу, таким образом, организуется поиск семи меньших, по сравнению с другими, элементов. Составление подобного кластера в памяти устройства представляет собой решение задачи нахождения координат опорных элементов.
- Следующий этап – преобразование информационного кадра, обеспечивает обработку считанной на предыдущем шаге информации о координатах середин проекций светящихся областей, на основе которой, в дальнейшем, будет сформировано сообщение в виде группы передаваемых полубайт, за счет использования маркера. Характерная черта итерации – большие вычислительные затраты.
- Вывод завершающего сообщение кадра. Здесь на первый план выходит сохранение целостности передаваемых сообщений. Для решения этого вопроса, как правило, применяют следующие меры: соблюдают строгую очередность выводимых кадров сообщения, а также отделяют группы кадров передаваемого сообщения от предыдущего и последующего сообщений. В данном случае большое значение имеет завершающий кадр, для формирования которого отлично подходит центральный синхронизирующий элемент, задействованный в распознавании каждого полубайта. Следовательно, погасший центральный элемент говорит об окончании передачи сообщения.

Реализация оптического маркера

Основными элементами системы передачи данных между мобильными роботами с помощью оптического маркера выступают: плата с микроконтроллером, модуль оптической передачи (дисплей) и приемная камера.

В микроконтроллере платы храниться программа вывода на экран дисплея данных в виде высвечиваемого оптическим маркером кода, а также программа обработки получаемого камерой изображения оптического маркера.

На рисунке 3 отображена структурная схема системы передачи/ приема данных от рассматриваемого оптического маркера.

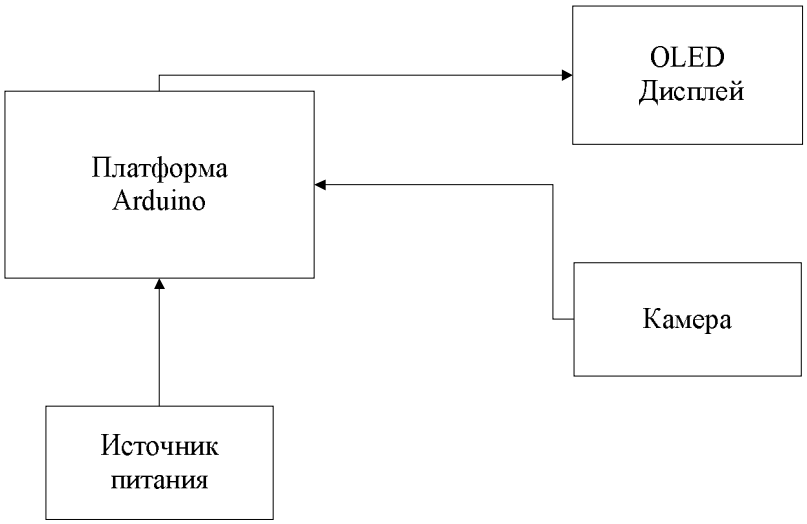


Рис. 3. Структурная схема системы передачи и приема данных от оптического маркера

В качестве платы используется отладочная плата на базе Arduino Uno, обладающая следующими достоинствами: малые габаритные размеры, низкая себестоимость, универсальный язык программирования (C++), наличие 14 цифровых и 6 аналоговых входов/выходов для подключения одного или нескольких устройств [3]. Микроконтроллер данной платы уже прошит загрузчиком, что не требует дополнительных программаторов и позволяет загружать код напрямую с компьютера.

Дисплей – элемент, на экране которого непосредственно будет отображаться оптический маркер, то есть дисплей является передающим модулем данной системы.

Для отображения предлагаемого оптического маркера будем использовать дисплей технологии OLED. Данный вид дисплея имеет ряд преимуществ по сравнению с графическими ЖК дисплеями. В ЖК технологии существует подсветка, а в технологии OLED в ней нет необходимости, так как каждый пиксель изображения будет светиться сам. Отсюда вытекает главное преимущество таких дисплеев – яркость и контрастность изображения. Ещё одним достоинством OLED дисплея являются большие углы обзора, то есть изображение без потери качества видно до 160 градусов, что удовлетворяет заданным характеристикам.

Применение готового дисплея более целесообразно, чем сборка оптического маркера с помощью отдельных светодиодов, для работы которых понадобятся сдвиговые регистры. Тем самым конструкция оптического маркера увеличится в размерах, а самое главное в случае «выгорания» какого-либо диода, конструкция перестанет правильно работать, чего не случится с использованием дисплея OLED. Во-первых, дисплей напрямую подключается к плате и не требует дополнительных элементов, во-вторых, каждый элемент будет подсвечивать группой диодов дисплея матрицы, а это значит при «выгорании» одного или нескольких диодов, оптический маркер продолжит исправно работать, так как происходит анализ всей области светящихся элементов оптического маркера.

Приемный модуль представляет собой *камеру*, которая будет считывать изображение маркера. Выбор остановим на Arduino камере, которая обладает возможностями по настройке насыщенности цветов и диафрагме, что позволяет получить кадры с достаточно высоким качеством, также она может использоваться для съёмки видео или отслеживания движения, что говорит о том, что данная камера обладает авто-контрастностью и авто-яркостью. Фотоматрица данной камеры технологии КМОП и выдает до 30 кадров в секунду.

Источником питания может служить: компьютер, если данная система подключена к нему через USB провод, например, для проверки работы данной системы; батарейка 9В для непосредственной работы в мобильном роботе. У дисплея и камеры рабочее напряжение со-

ставляет 5В, данное напряжение подается им от платы Arduino, которая имеет в своем распоряжении соответствующий рабочий выход.

Вывод

В данной статье, на примере системы беспроводной передачи данных для коллектива мобильных роботов, был описан один из возможных способов повышения надежности передачи сообщений по оптическим беспроводным каналам за счет использования маркеров с улучшенным распознаванием элементов. Показано, что данный маркер упрощает восприятие передаваемой информации и повышает достоверность передачи на заданном расстоянии.

Библиографический список

1. Аверина Л. И. Системы цифровой связи: учебное пособие /Л. И. Аверина. — Воронеж: ВГУ, 2016. — 50 с.

2. Ефимов А. И., Колчаев Д. А., Логинов А. А. [и др.]; под редакцией А. В. Воробьёва, М. Б. Никифорова Ввод–вывод изображений в авиационных системах технического зрения: учебное пособие / А. И. Ефимов, Д. А. Колчаев, А. А. Логинов [и др.]; под редакцией А. В. Воробьёва, М. Б. Никифорова. — Москва: ФИЗМАТЛИТ, 2020. — 248 с.

3. Петин В. В. 77 проектов для Arduino: учебно-методическое пособие / В. В. Петин. — Москва: ДМК Пресс, 2020. — 356 с

СЕКЦИЯ «МОДЕЛИ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В САПР»

УДК 681.532.8; ГРНТИ 50.43.15

ПОДАВЛЕНИЕ АВТОКОЛЕБАНИЙ ПРИ ПОМОЩИ ШИРОТНО-ИМПУЛЬСНОЙ МОДУЛЯЦИИ УПРАВЛЯЮЩИХ ИМПУЛЬСОВ В НЕЛИНЕЙНЫХ СИСТЕМАХ АВТОМАТИЧЕСКОГО РЕГУЛИРОВАНИЯ

А.Н. Кащеев*, В.И. Хрюкин**

*Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, *lexakasheev@yandex.ru, **vi_x@mail.ru*

Аннотация. В данной работе рассматривается метод решения проблемы автоколебаний при помощи широтно-импульсной модуляции управляющего сигнала.

Ключевые слова: широтно-импульсная модуляция (ШИМ), автоколебания, нелинейные системы.

SUPPRESSION OF SELF-OSCILLATIONS BY MEANS OF PULSE–WIDTH MODULATION OF CONTROL PULSES IN NONLINEAR AUTOMATIC CONTROL SYSTEMS

A.N. Kashcheev*, V.I. Khryukin**

*Ryazan State Radio Engineering University named after V.F. Utkin, Russian Federation, Ryazan, *lexakasheev@yandex.ru, **vi_x@mail.ru*

Annotation. In this paper, we consider a method for solving the problem of self-oscillation using pulse-width modulation of the control signal.

Keywords: pulse width modulation (PWM), self-oscillation, nonlinear systems.

При исследовании систем автоматического регулирования (САР) стремятся линеаризовать исходную систему, так как методы анализа и синтеза линейных систем наиболее полно разработаны. Но в большинстве технических систем отдельные звенья невозможно линеаризовать традиционными методами путем разложения в ряд Тейлора и отбрасывания нелинейных членов разложения [3]. Так САР, где в качестве устройства управления используется микроконтроллеры без цифро-аналогового преобразователя (ЦАП) или ЦАП с ограниченным числом разрядов, являются существенно нелинейными. В этом случае в САР выделяют линейную и нелинейную части и используют специальные методы, например, метод гармонической линеаризации [2]. Однако такие системы в отличии от линейных, имеют еще одно состояние (режим) – «автоколебаний». Автоколебания - это устойчивые собственные колебания, возникающие за счет непериодического источника энергии и определяемыми свойствами системы [2]. В линейной системе при малейшем изменении параметров колебательный процесс становится либо затухающим, либо расходящимся. Автоколебания же являются устойчивым режимом: малые изменения параметров системы не выводят ее из этого режима. Автоколебания в нелинейных системах в общем случае нежелательны, а иногда не допустимы.

Эффективным способом подавления автоколебаний является наложение на релейный элемент вынужденных колебаний достаточно высокой частоты по сравнению с частотой самих автоколебаний. Такой метод подавления называют вибрационной линеаризацией нелинейной характеристики релейного элемента [1]. При вибрационной линеаризации обеспечивается плавная зависимость среднего значения (постоянной составляющей) выходного напряжения реле от значения медленно меняющейся входной величины.

Другим способом устранения автоколебаний или уменьшения их амплитуды заключается в деформации логарифмических частотных характеристик линейной части системы [1]. Автоколебания можно устранить, если добиться, чтобы логарифмическая фаза-частотная характеристика (ЛФЧХ) линейной части системы 1 не пересекала кривую 2. Вид деформированной ЛФЧХ показан на рис.1 пунктирной кривой 1. Для уменьшения амплиту-

ды автоколебаний нужно стремиться к тому, чтобы кривая 3 пересекала фазовую границу устойчивости (ФГУ) в точке, соответствующей меньшим значениям относительной амплитуды автоколебаний. Для уменьшения амплитуды автоколебаний обычно следует перемещать точку пересечения характеристик в область более высоких частот, например, из точки В в С (рис.1).

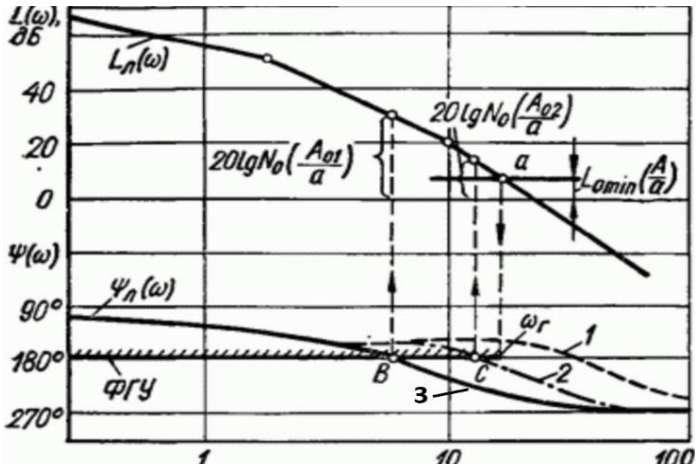


Рис. 1. ЛАЧХ и ЛФЧХ

Желаемая деформация частотных характеристик линейной части системы осуществляется обычно с помощью линейных или нелинейных корректирующих устройств.

Еще один известный метод подавления автоколебаний в системах с цифровым управлением является использование цифро-аналоговый преобразователь (ЦАП). По сути ЦАП это тоже нелинейное звено как и реле, однако оно может формировать на выходе 2^N - уровней напряжения (где N – разрядность ЦАП). Чем больше разрядность ЦАП, тем точнее можно задать управляющий сигнал цифровой управляющей машины (ЦУМ), и тем меньше будет амплитуда автоколебаний. Однако применение ЦАП имеет следующие особенности: не во всех моделях микроконтроллеров имеется встроенный ЦАП, ЦАП должен обладать достаточной разрядность и быстродействием, применение ЦАП с большой разрядностью требует задействовать соответствующее число портов ввода-вывода микроконтроллера и др.

В настоящей статье предлагается решить проблему автоколебаний при помощи широтно-импульсной модуляции (ШИМ) управляющего сигнала. Рассмотрим это на примере САР «Испытательный стенд»:

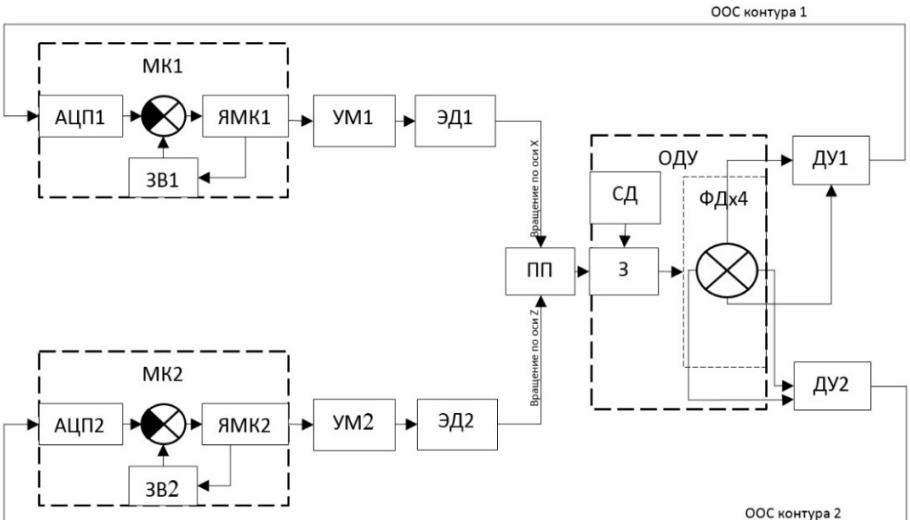


Рис. 2. Функциональная блок-схема испытательного стенда

На рисунке 2 приняты следующие сокращения: МК1,2 – микроконтроллер; АЦП1,2 – аналого-цифровой преобразователь; ЯМК1,2 – ядро микроконтроллера (процессор); ЗВ1,2 – задающее воздействие; УМ1,2– усилитель мощности; ЭД1,2 – электродвигатель; ПП – подвижная платформа; ОДУ – оптический датчик угла; З – зеркало; СД – светодиод; ФДх4 – четырех сегментный фотодатчик; ДУ1,2 – дифференциальный усилитель. Такая САР работает следующим образом: микроконтроллер МК1 формирует задающее воздействие ЗВ1, из которого вычитается оцифрованное на АЦП1 значение потенциала с ОДУ. Далее полученное значение (ошибка регулирования) обрабатывается в ЯМК1, согласно заложенному в него алгоритму управления. Затем, в зависимости от значения вычисленной управляющей величины, на выходе МК1 формируется управляющий сигнал в +5В или -5В, который усиливается по мощности в УМ1. Далее это сигнал поступает на электродвигатель ЭД1 в результате чего на его валу возникает момент силы, который поворачивает подвижную платформу ПП вместе с зеркалом З вокруг оси X. Светодиод СД, под действием постоянного тока, формирует световой поток, который отражаясь от зеркала З с определенной степенью освещает четырех сегментный фотодатчик ФДх4 (степень освещенности каждого из сегментов зависит от угла рассогласования зеркала и фотодатчика). Пусть свет преимущественно падает на сегментную пару, отвечающий за ориентацию подвижной платформы ПП по оси X (вертикальные сегменты ФДх4), тогда на соответствующих выводах ФДх4 формируются два электрических сигнала, которые поступают на входы дифференциального усилителя ДУ1, который определяет разность этих сигналов. Далее сигнал разности поступает на вход ЦАП1 МК1, тем самым замыкая отрицательную обратную связь (ООС). Аналогичным образом работает второй контур по регулированию оси Z.

Передаточные функции звеньев представленной системы следующие:

$$W_{ЦУМ} = \frac{76{,}66z - 68{,}93}{z - 0{,}229}; \quad (1)$$

$$W_{УМ} = 1; \quad (2)$$

$$W_{ЭД} = \frac{0{,}003}{0{,}0005s + 1}; \quad (3)$$

$$W_{ПП} = \frac{312{,}5}{0{,}1756s^2 + s}; \quad (4)$$

$$W_{ОДУ} = 12. \quad (5)$$

Так как два контура управления испытательного стенда имеют одинаковый принцип работы, то для простоты будем рассматривать только один контур. На следующем рисунке приведена блок-схема одного контура управления испытательного стенда разработанная с использованием ППП Matlab Simulink:

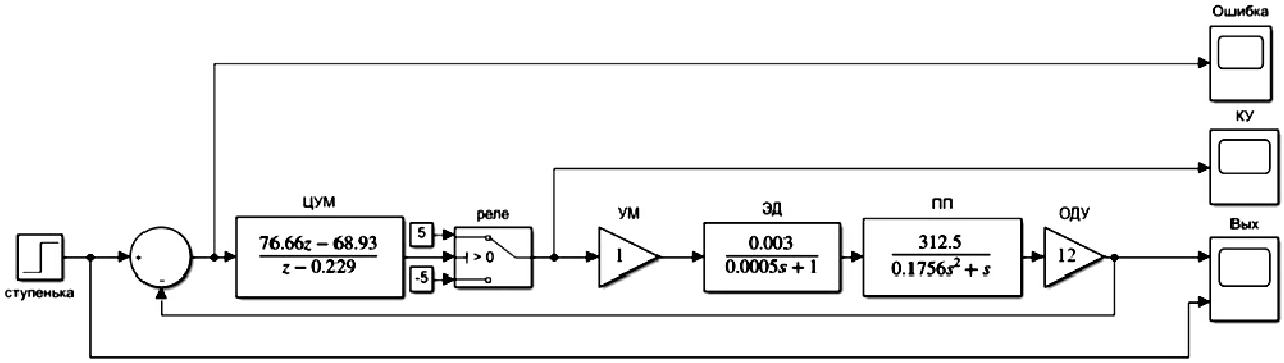


Рис. 3. Блок-схема одного контура управления испытательного стенда

Моделирование представленной системы дает следующий результат:

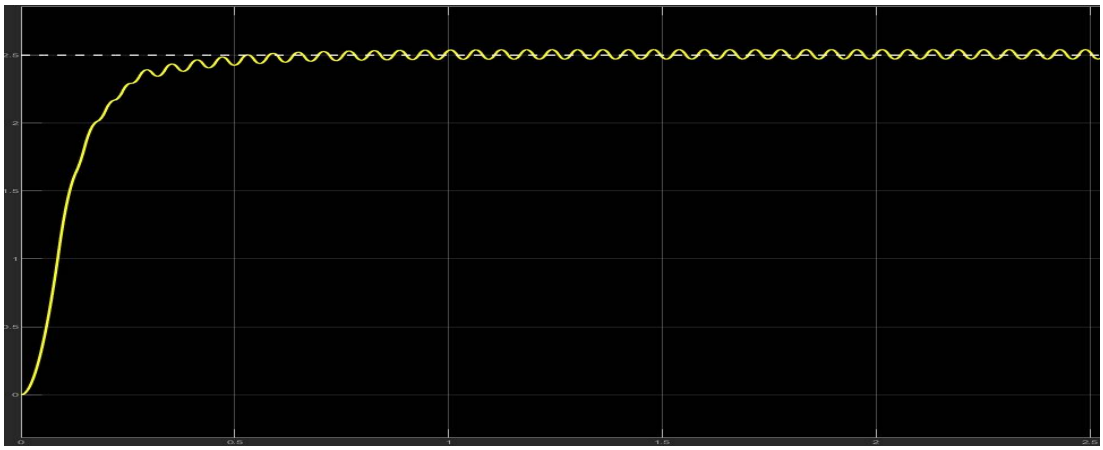


Рис. 4. Диаграмма выходной величины напряжения

Из диаграммы видно, что переходной процесс протекает нормально, однако, когда регулируемая величина приближается к заданному значению, возникает автоколебание. Это происходит из-за того, что алгоритм ЦУМ не учитывает релейное звено: ЦУМ формирует управляющий сигнал некоторой величины, который обеспечивает качественность протекания переходного процесса, однако реле на выходе преобразует этот сигнал в одно из двух значений +5В или -5В (рис. 5а), в результате чего возникает перерегулирование контролируемой величины, на которое система также импульсивно реагирует.

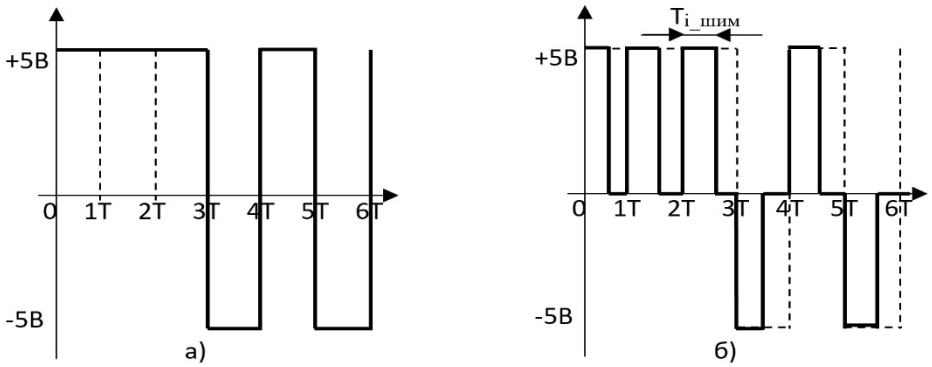


Рис. 5 – а) управляющий сигнал (двухпозиционное реле);
б) управляющий сигнал с ШИМ (трехпозиционное реле)

Поэтому, чтобы уменьшить амплитуду автоколебаний предлагается изменять длительность управляющего импульса, то есть производить широтно-импульсную модуляцию (рис.5б). Здесь же возникает вопрос – какую скважность моделировать управляющим импульсам для обеспечения качества работы САР? Если уменьшать длительность импульса, то амплитуда автоколебаний будет уменьшаться, однако вместе с ней будет расти длительность переходного процесса (ухудшается быстродействие САР). По этой причине предлагается использовать ШИМ управляющего воздействия только тогда, когда регулируемая величина приближается к заданному значению. Тогда алгоритм работы такой САР следующий: если абсолютная величина ошибки регулирования «x» больше некоторого значения «ε» то выполнять алгоритм управления ЦУМ, в противном случае также выполнять алгоритм управления ЦУМ но с использованием широтно-импульсной модуляцией управляющих импульсов по следующей формуле:

$$T_{ШИМ} = T\frac{|x|}{\varepsilon}, \tag{6}$$

где T – время дискретизации ЦУМ;
x – ошибка регулирования;
ε - порог срабатывания ШИМ.

Время дискретизации T можно задать при помощи таймера микроконтроллера, а в качестве значения параметра ε предлагается принять амплитуду автоколебаний.

Согласно выше описанному алгоритму модель САР примет следующий вид:

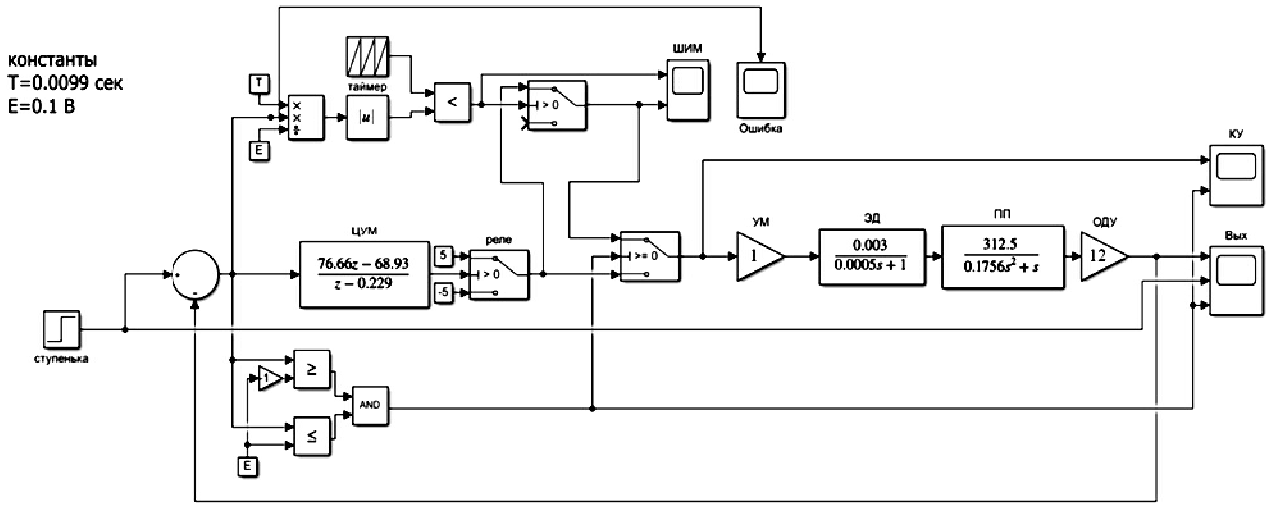


Рис. 6. Модифицированная блок-схема испытательного стенда

Диаграммы работы такой системы представлены на рисунках 7 (выходная величина с ШИМ управляющего воздействия) и 8 (диаграмма работы ЦУМ).

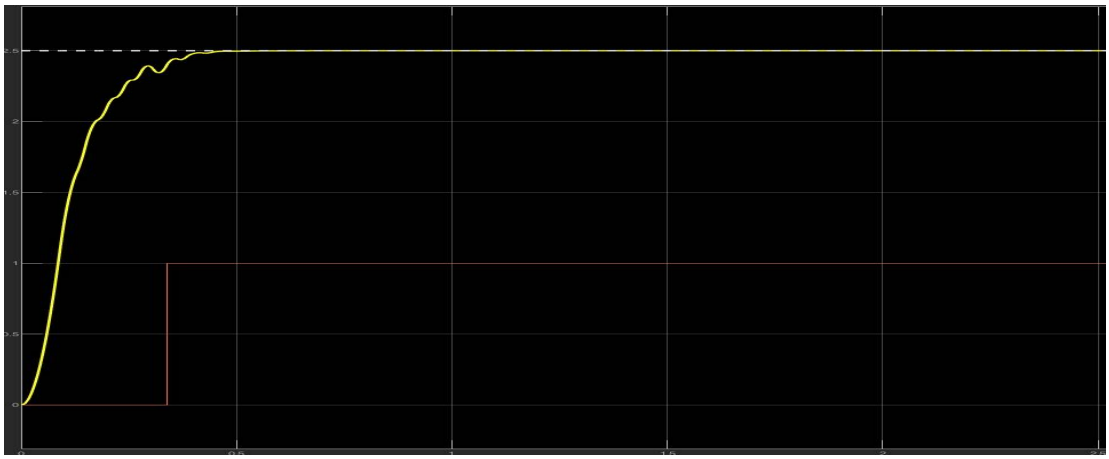


Рис. 7. Диаграмма выходной величины с ШИМ управляющего воздействия

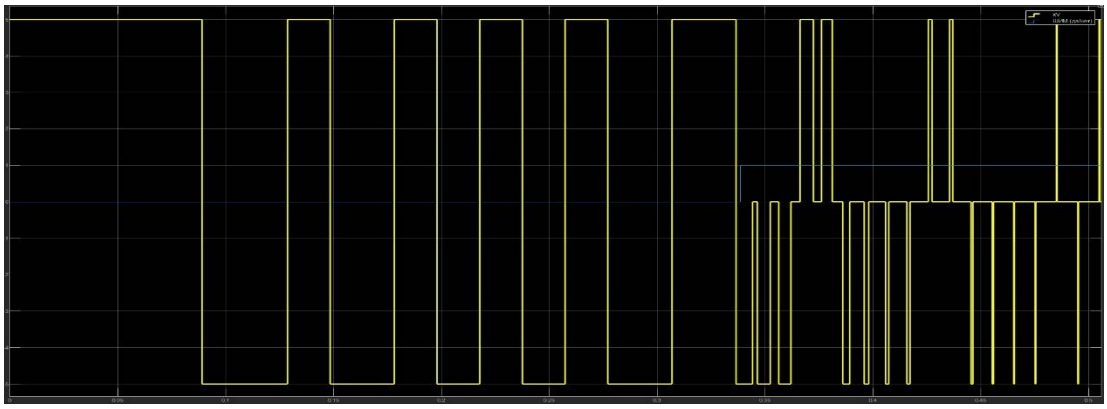


Рис. 8. Диаграмма работы ЦУМ (часть диаграммы)

На практике предложенный метод реализуется программным способом с использованием таймера микроконтроллера. Таймер следует настроить на режим CTC (Clear Time on Compare - сброс при совпадении), который формирует прерывание при достижении счетного регистра значения регистра сравнения. Значение регистра сравнения должно быть эквивалентно значению $T_{ШИМ}$, то есть должен быть подобран корректный делитель тактовых импульсов. В обработчике (функции) прерывания следует выполнить сброс всех портов, которые учавствуют в управлении электродвигателем. Таким образом, можно осуществить ШИМ управляющего сигнала для подавления автоколебаний.

Биографический список

1. Зайцев Г. Ф. Теория автоматического управления и регулирования. — 2-е изд., перераб, и доп.— К.: Выща шк. Головное изд-во, 1989. — 431 с.

2. Лазарева Т. Я., Мартеньяков Ю. Ф. Основы теории автоматического управления: Учебное пособие. 2-е изд., перераб. И доп. Тамбов: Изд-во Тамб. Гос. Ун-та, 2004. 352 с.

3. Хрюкин В. И. Основы управления техническими системами: учеб. пособие: Рязан. гос. радиотех. ун-т. Рязань, 2022. 112 с.

УДК 681.51; ГРНТИ 50.43.17

РАЗРАБОТКА АЛГОРИТМА ФУНКЦИОНИРОВАНИЯ МИКРОПРОЦЕССОРНОЙ СИСТЕМЫ УПРАВЛЕНИЯ ТЕХНИЧЕСКИМ ОБЪЕКТОМ В РЕАЛЬНОМ ВРЕМЕНИ

С.В. Скворцов, В.И. Хрюкин

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, VI_X@mail.ru

Аннотация. В работе рассмотрены вопросы разработки алгоритма формирования управляющего кода для автоматической системы на базе микроконтроллера. Показано применение среды MatLab/Simulink для анализа устойчивости и качества цифровой системы управления, реализующей разработанный алгоритм.

Ключевые слова: микропроцессорная система управления, микроконтроллер, передаточная функция, алгоритм формирования управляющего кода, моделирование цифровой системы управления.

DEVELOPMENT OF MICROPROCESSOR CONTROL SYSTEM FUNCTIONING ALGORITHM FOR A TECHNICAL OBJECT IN REAL TIME

S.V. Skvortsov, V.I. Khryukin

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, VI_X@mail.ru

Abstract. The paper considers the issues of developing an algorithm for generating control code for an automatic system based on a microcontroller. The application of the MatLab/Simulink environment to analyze the stability and quality of a digital control system implementing the developed algorithm is shown.

Keywords: microprocessor control system, microcontroller, transfer function, algorithm for generating the control code, simulation of a digital control system.

Управление сложными техническими объектами (ТО) в режиме реального времени является одной из важнейших областей применения микропроцессорной техники и микроконтроллеров [1]. Для обеспечения заданного качества процессов управления в работе [2] предложена модель системы на базе микроконтроллера (МК), которая реализует принцип управления по ошибке [3] и строится по схеме, показанной на рисунке 1.

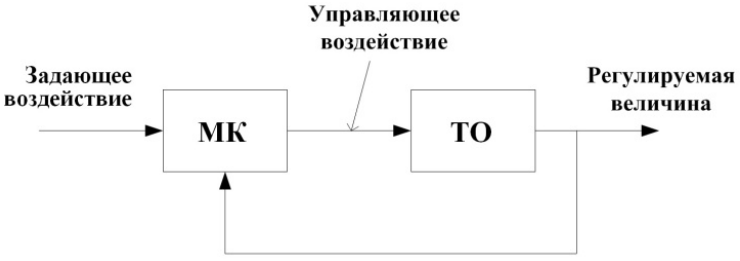


Рис. 1. Модель системы с обратной связью

Структурная схема системы управления на базе МК представлена на рис. 2, где $g(t)$ – входная величина (задающее воздействие); $x(t)$ – ошибка регулирования; $x_1(t)$ – управляющее (регулирующее) воздействие; $x_2(t)$ – регулируемая величина; $D(z)$ – передаточная функция, реализуемая в виде алгоритма функционирования МК; $W(s)$ – передаточная функция ТО.

Динамику большинства технических объектов с точностью, обеспечивающей практическое применение, можно описать дифференциальным уравнением третьего порядка:

$$c_0 \dddot{x}_2(t) + c_1 \ddot{x}_2(t) + c_2 \dot{x}_2(t) + c_3 x_2(t) = d_0 x_1(t), \quad (1)$$

где c_0, c_1, c_2, c_3, d_0 – числовые коэффициенты, определяющие характеристики конкретного ТО.

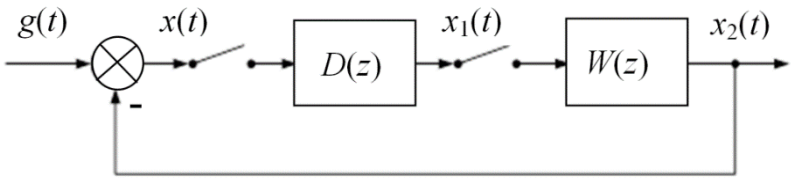


Рис. 2. Структурная схема микропроцессорной системы управления

Анализ передаточной функции

$$W_0(s) = \frac{d_0}{c_0 s^3 + c_1 s^2 + c_2 s + c_3}, \tag{2}$$

полученной из уравнения (1), где s – комплексная переменная в преобразовании Лапласа, частотными методами [3] позволяет оценить устойчивость системы управления. Если устойчивость не обеспечивается, выполняется коррекция сигнала ошибки путем формирования управляющего воздействия $x_1(t)$ на объект. Итерационное выражение, описывающее алгоритм управления для системы, показанной на рис. 2, с передаточной функцией (1) будет иметь следующий вид [4]:

$$x_1[nT] = \frac{\left(1+\frac{2\tau}{T}\right)x[nT] + \left(1-\frac{2\tau}{T}\right)x[(n-1)T]}{2}, \tag{3}$$

где T – время дискретизации, т.е. время выполнения алгоритма управления;

n – номер итерации;

τ - постоянная времени, определяющая динамику системы.

Алгоритм функционирования микропроцессорной системы управления

Соотношение (3) позволяет разработать алгоритм функционирования микропроцессорной системы управления, которая формирует регулирующее воздействие $x_1[nT]$ в момент времени $t = nT$ (n = 1, 2, …). Исходными данными являются значения $x[nT]$, $x[(n\text{-}1)T]$ сигнала ошибки регулирования, получаемые от датчиков положения, в текущий nT и предыдущий $(n\text{-}1)T$ моменты времени. Итерационное выражение (3) позволяет определить алгоритм работы микроконтроллера в режиме реального времени, который показан на рис. 3.

Для разработки программного обеспечения уравнение (3) представляется в эквивалентном виде:

$$x_1[nT] = \frac{b_0 x[(n-1)T] + b_1 x[nT]}{2}, \tag{4}$$

которая показывает, что в алгоритме управления будут использоваться операции умножения прямого кода сигнала ошибки $x[(n\text{-}1)T]$, $x[nT]$ на вещественные коэффициенты $b_0/2$, $b_1/2$ и сложения чисел со знаком. Формирование кода управляющего воздействия выполняется по алгоритму, представленному на рисунке 4.

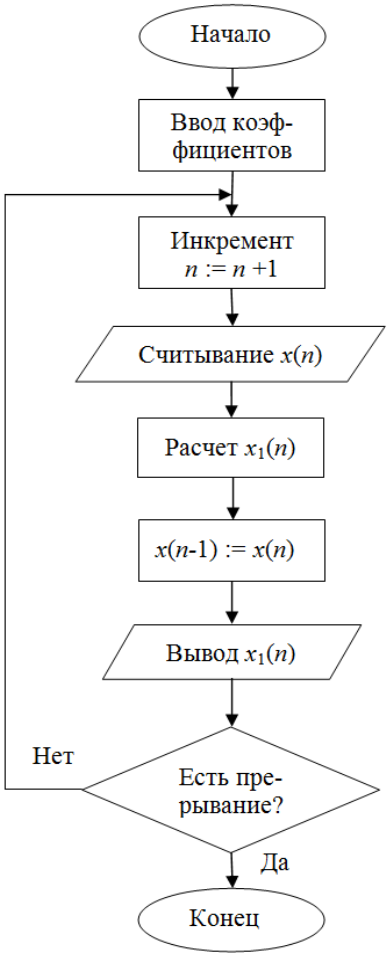


Рис. 3. Обобщенный алгоритм управления

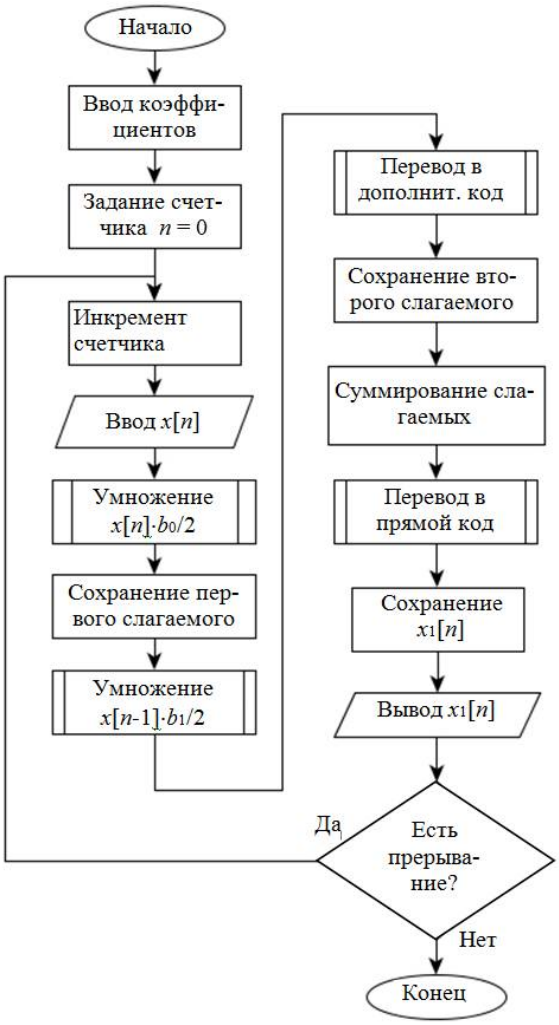


Рис. 4. Алгоритм формирования кода управляющего воздействия

При программной реализации алгоритма формирования кода управляющего воздействия на машинно-ориентированном языке необходимо учитывать особенности системы команд микроконтроллера [5, 6]. В частности, алгебраическое сложение чисел должно выполняться с использованием дополнительного кода.

Умножения целого числа со знаком на вещественный коэффициент можно реализовать с помощью последовательности операций, указанных в работе [2].

При умножении длина числового кода результата удваивается относительно длины исходных данных. В связи с этим требуется выполнить сложение чисел, разрядность которых превышает машинное слово. Суммирование данных большой разрядности (длина которых больше одного байта) производится следующим образом:

1) сформировать дополнительный код отрицательных чисел;

2) просуммировать младшие байты операндов;

3) добавить к старшему байту одного из операндов бит переноса от предыдущей операции;

4) прибавить к результату значение старшего байта другого операнда.

Моделирование динамики микропроцессорной системы управления

Проверка устойчивости и качества системы управления может быть выполнена путем моделирования ее динамики в среде MatLab/Simulink. Модель исследуемой системы управления на базе микроконтроллера показана на рис. 4, где b_0, b_1 – параметры передаточной функции, определяемые уравнением (4).

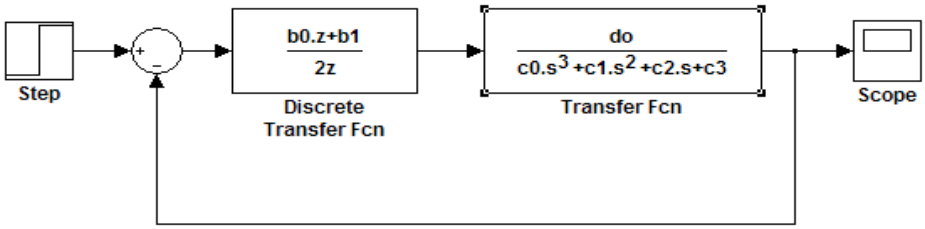


Рис. 4. Модель алгоритма функционирования микроконтроллера в Simulink

Оценку качества системы управления на базе микроконтроллера можно выполнить на основе анализа временной характеристики (рис. 5), которая является следствием воздействия единичного ступенчатого сигнала на вход модели (рис. 4).

Пример переходного процесса, полученного для САР с передаточными функциями

$$W_0(s) = \frac{218}{0{,}0035s^3 + 1{,}52s^2 + 0{,}157s}, \quad D(z) = \frac{11{,}8z - 9{,}8}{2z},$$

представлен на рис. 5.

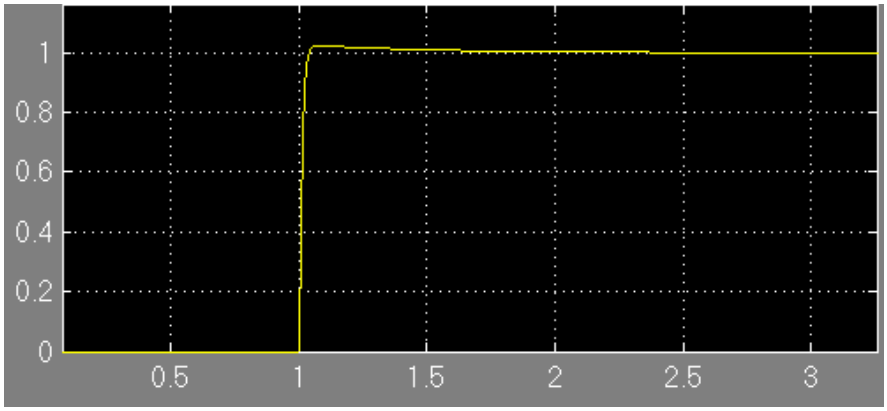


Рис. 5. Переходный процесс САР

Величина перерегулирования определяется по формуле

$$\sigma = \frac{h_{\max} - h_{\text{уст}}}{h_{\text{уст}}},$$

где $h_{\max}$ – максимальное значение переходной характеристики;

$h_{\text{уст}}$ – установившееся значение переходной характеристики.

Например, на рисунке 5 показан случай, когда перерегулирование σ составляет около двух процентов. Заметим, что для качественных систем эта величина не должна превышать 30 %.

Такая временная характеристика системы также позволяет определить длительность переходного процесса, т.е. момент времени, когда выполняется условие

$$| h(t) - h_{уст}| \leq \Delta,$$

где $\Delta = (0{,}01 \ldots 0{,}05)\, h_{уст}$.

Для указанного примера эта величина на уровне 1 % составит примерно 0,175 сек. (рис. 6).

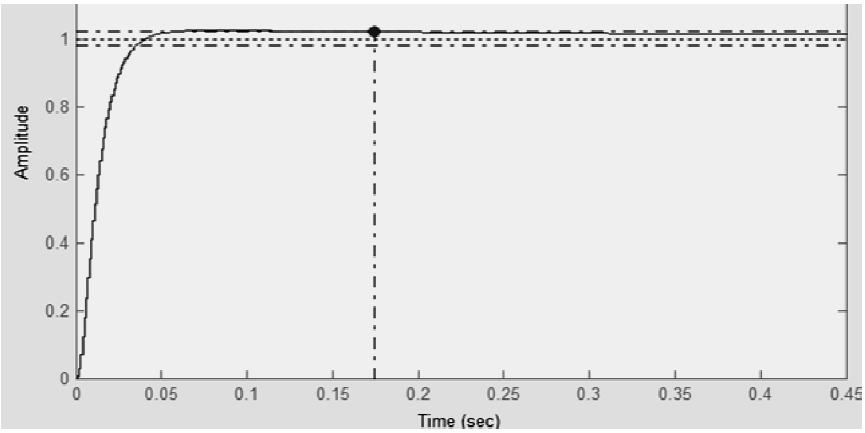


Рис. 6. Переходная характеристика

Таким образом, результаты моделирования позволяют сделать вывод о возможности построения микропроцессорной системы, реализующей разработанный алгоритм управления.

Выводы

Предложенная методика позволяет разработать для автоматической системы на основе микроконтроллера алгоритм управления, обеспечивающий устойчивость и качество такой цифровой системы. Представлены схемы алгоритмов функционирования микропроцессорной системы управления и формирования кода управляющего воздействия, предназначенные для реализации на машинно-ориентированном языке программирования.

Библиографический список

1. Микропроцессорные системы автоматического управления / под ред. В.А. Бесекерского. Л.: Машиностроение, 1988. 365 с.

2. Микропроцессорная система управления технологическим процессом: методические указания к курсовому проектированию / Рязан. гос. радиотехн. ун-т; С.В. Скворцов, В.И. Хрюкин. Рязань, 2016. 28 с.

3. Хрюкин В.И. Основы управления техническими системами: учеб. пособие / Рязан. гос. радиотехн. ун-т. Рязань, 2022. 112 с.

4. Скворцов С.В., Хрюкин В.И. Модель микропроцессорной системы управления техническим объектом в режиме реального времени // Материалы VII Всероссийской научно-технической конференции «Актуальные проблемы современной науки и производства». Рязань: РГРТУ, 2022. С. 282-289.

5. Скворцов С.В., Хрюкин В.И. Организация микропроцессоров и микропроцессорных систем: учеб. пособие / Рязан. гос. радиотехн. ун-т. Рязань, 2018. 80 с.

6. Гуров В.В. Микропроцессорные системы. М.: ИНФРА-М, 2016. 368 с.

УДК 004.72; ГРНТИ 50.41.23

ПОСТАНОВКА ЗАДАЧИ ПОСТРОЕНИЯ СИСТЕМЫ НЕЧЕТКОГО ВЫВОДА ДЛЯ МНОГОКРИТЕРИАЛЬНОГО АЛГОРИТМА ПАРНЫХ ПЕРЕХОДОВ В ПРОГРАММНО-КОНФИГУРИРУЕМЫХ СЕТЯХ

Д.А. Перепелкин, К.В. Анисимов
Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, perepelkin.d.a@rsreu.ru

Аннотация. В работе рассматриваются достоинства и недостатки алгоритмов маршрутизации, описывается работа алгоритма парных переходов каналов связи программно-конфигурируемой сети. Формулируется задача построения системы нечеткого вывода для многокритериального алгоритма парных переходов.
Ключевые слова: программно-конфигурируемые сети, алгоритмы маршрутизации, система нечеткого вывода.

FORMULATION OF THE PROBLEM OF CONSTRUCTING A FUZZY INFERENCE SYSTEM FOR A MULTICRITERIA ALGORITHM OF PAIR TRANSITIONS OF COMMUNICATION CHANNELS IN SOFTWARE-DEFINED NETWORKS

D.A. Perepelkin, K.V. Anisimov
Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, perepelkin.d.a@rsreu.ru

The summary. The paper discusses the advantages and disadvantages of routing algorithms, describes the operation of the algorithm for pair transitions of communication channels in a software-defined network. The problem of constructing a fuzzy inference system for a multicriteria algorithm of pair transitions is formulated.
Keywords: software-configurable networks, routing algorithms, fuzzy output system.

В настоящее время существует множество алгоритмов, предназначенных для построения маршрутов между узлами сети. Наиболее распространенными являются алгоритмы Дейкстры, Бэллмана – Форда, Флойда – Уоршелла. Все они имеют как преимущества, так и недостатки. Так, алгоритм Дейксры имеет временную сложность $O(N^2)$, а алгоритм Бэллмана – Форда – $O(N^3)$, однако он допускает наличие в сети ребер отрицательного веса. Алгоритм Флойда – Уоршелла имеет сложность $O(N^3)$ и находит пути между всеми парами узлов сети. Большим недостатком этих алгоритмов является то, что при изменении сетевой метрики одного канала связи для поддержания необходимого качества сервиса приходится пересчитывать все найденные маршруты и производить перестроение таблиц потоков в OpenFlow коммутаторах.

Этого недостатка лишен алгоритм парных переходов каналов связи [1].

Алгоритм парных переходов каналов связи

Алгоритм парных переходов каналов связи ПКС позволяет обеспечивать оптимальную маршрутизацию в условиях динамически изменяющихся параметров каналов связи. Работа алгоритма основана на разделении всего множества каналов связи на два подмножества T_N и T_R. Первое – подмножество каналов, входящих в дерево оптимальных маршрутов от некоторого начального узла S_0 до всех остальных. Дерево строится с помощью алгоритма Дейкстры. Второе – подмножество каналов, не вошедших в дерево оптимальных маршрутов. При возникновении некоторых условий канал $\{S_i, S_j\} \in T_N$ может перейти из T_N в T_R, а парный ему канал $\{S_k, S_j\} \in T_R$ наоборот перейдет из T_R в T_N.

При работе алгоритма решаются две задачи: определение условий для осуществления парных переходов и определение парных каналов.

Пусть имеется сеть, в которой уже построено дерево оптимальных маршрутов с корневым узлом S_0. Путь от S_0 до каждого узла S_i характеризуется оценкой $d_{0,i}$. Рассмотрим произвольный узел с номером k, родительским узлом которого в дереве оптимальных маршрутов является узел с номером p (рис.).

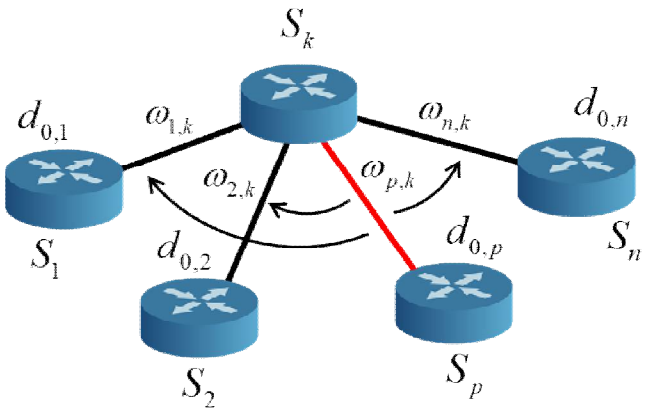


Рис. Определение парного канала связи

В случае, если метрика канала $\{S_p, S_k\}$ $\omega_{p,k}$ превысит значение $\omega^s_{p,k}$, называемое точкой выхода из дерева оптимальных маршрутов, то $\{S_p, S_k\}$ перейдет из T_N в T_R, а парный ему канал $\{S_{pr}, S_k\}$ перейдет из T_R в T_N. Значение точки выхода из дерева определяется следующим образом:

$$\omega^s_{p,k} = \min_{i \in N}(d_{0,i} + \omega_{i,k} - d_{0,p}),$$

где i – номер некоторого узла, смежного с S_k;

N – множество номеров узлов, смежных с S_k.

При этом, номер парного узла определяется как: $pr = \arg(\omega^s_{p,k})$.

В случае, если метрика канала $\{S_i, S_k\}$ $\omega_{i,k}$ станет меньше значения $\omega^t_{i,k}$, называемого точкой вхождения в дерево оптимальных маршрутов, то $\{S_i, S_k\}$ перейдет из T_R в T_N, а канал $\{S_p, S_k\}$ перейдет из T_N в T_R. Значение точки вхождения в дерево для канала $\{S_i, S_k\}$ определяется как: $\omega^t_{i,k} = d_{0,k} - d_{0,i}$.

Таким образом, используя алгоритм парных переходов, при изменении метрики канала связи нет необходимости пересчитывать все маршруты. Достаточно совершить парные переходы и пересчитать точки вхождения в дерево оптимальных маршрутов и выхода из него только для тех каналов, которые находятся ниже по иерархии.

Многокритериальный алгоритм парных переходов каналов связи

Реальные каналы связи имеют не одну, а множество сетевых метрик. В качестве таковых могут выступать: задержка, пропускная способность, процент потерь пакетов и т.д. Задача маршрутизации сети, каналы связи которой имеют несколько метрик, значительно

усложняется. При применении алгоритма парных переходов к такой сети, возникает два вида неопределенностей. Первая связана с тем, что нет четкого понимания, когда производить парный переход. Вторая связана с тем, что невозможно однозначно определить парные каналы связи, ведь между хорошим и плохим качеством сервиса [2] нет четкой границы. Например, один канал передачи данных может иметь низкий процент потерь пакетов, но высокую задержку. Другой – высокий процент потерь пакетов, но низкую задержку. Качество сервиса этих каналов будет примерно одинаковое так как при потере пакетов при передаче по второму каналу их будет можно быстро отправить повторно.

Для работы в условиях неопределенности хорошо подходит аппарат теории нечетких множеств и нечеткой логики. Системы нечеткого [3] вывода обладают интерполяционными свойствами и позволяют аппроксимировать функции нескольких переменных.

Построение системы нечеткого вывода состоит из следующих шагов.

1. Определение входных лингвистических переменных. Ими могут являться значения метрик каналов связи или их приращения.

2. Определение выходных лингвистических переменных.

3. Описание входных и выходных лингвистических переменных с помощью терм множеств. Для этого можно использовать различные функции принадлежности [4].

4. Создание базы продукционных правил. Это наиболее трудоемкий процесс при построении системы нечеткого вывода, ведь на этом этапе необходимо учитывать мнение разных экспертов данной предметной области, а число правил определяется произведением количества терм-множеств каждой входной лингвистической переменной.

5. Выбор метода дефаззификации.

Формулируемая задача звучит следующим образом: построить систему нечеткого вывода, которая позволяла бы определять оптимальные парные переходы в условиях многокритериальности метрик каналов связи программно-конфигурируемых сетей.

Работа выполнена при финансовой поддержке гранта Президента РФ МД-3201.2022.1.6.

Библиографический список

1. Корячко В. П., Перепелкин Д. А. Программно-конфигурируемые сети. Учебник для вузов. М.: Горячая линия-Телеком, 2020. 288 с.

2. Koryachko V. P., Perepelkin D. A., Ivanchikova M. A., Byshov V. S., Tsyganov I. Yu. Analysis QoS Metrics in Software Defined Networks. Proceedings MECO 2017 – IEEE 6th Mediterranean Conference on Embedded Computing (MECO-2017), 2017, pp. 374-378. DOI: 10.1109/MECO.2017.7977240.

3. Практикум по основам теории нечетких множеств и нечеткой логики. Часть 2. Нечеткая логика: учеб. пособие / К. В. Анисимов, А. Н. Конюхов, К. А. Ципоркова; Рязан. гос. радиотехн. ун-т. Рязань, 2022. 88 с.

4. Практикум по основам теории нечётких множеств и нечёткой логики. Часть 1. Нечёткие множества: учеб.-метод. пособие / К.В.Анисимов, А.Н.Конюхов; Рязан. гос. радиотехн. ун-т. Рязань, 2021. – 68 с.

УДК 004.896; ГРНТИ 50.51.17

СТРУКТУРА ПРОГРАММНОГО МОДУЛЯ АВТОМАТИЗИРОВАННОГО РАЗМЕЩЕНИЯ КОМПОНЕНТОВ НА ОСНОВЕ МУЛЬТИАГЕНТНЫХ СИСТЕМ

Д.А. Перепелкин, В.Ю. Ликучев

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, v.likuchov@yandex.ru

Аннотация. В работе представлена обобщенная структура интегрируемого программного модуля автоматизированного размещения компонентов на печатной плате, в основу построения и функционирования которого заложены модели и принципы решения данной задачи средствами мультиагентных систем. В работе дано обоснование для создания модуля и использования при его построении агентно-ориентированного подхода и мультиагентной модели. Приведена аксиоматика агентно-ориентированного подхода в решении задачи. Определены алгоритмы частных процедур: структурирования и непосредственного размещения. Описаны ключевые компоненты модуля, приведена схема его структуры.

Ключевые слова: автоматизация проектирования, топологическое проектирование, размещение элементов, мультиагентная система, электронный модуль, печатная плата.

THE STRUCTURE OF THE SOFTWARE MODULE FOR AUTOMATED PLACEMENT OF COMPONENTS BASED ON MULTI-AGENT SYSTEMS

D.A. Perepelkin, V.Y. Likuchev

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, v.likuchov@yandex.ru

The summary. The article presents a generalized structure of an integrated software module for automated placement of components on a printed circuit board, the basis for the construction and operation of which are models and principles for solving this problem using multi-agent systems. The article provides a rationale for creating a module and using an agent-oriented approach and a multi-agent model in its construction. The axiomatics of the agent-oriented approach in solving the problem is given. Algorithms of private procedures are defined: structuring and direct placement. The key components of the module are described, a diagram of its structure is given.

Keywords: design automation, topological design, placement of elements, multi-agent system, electronic module, printed circuit board.

Введение

Высокая плотность размещения компонентов на печатной плате создает большие трудности при реализации соединений между ними. В этой связи задача размещения на плоскости определяет быстроту и качество трассировки. Оптимальное размещение компонентов обеспечивает повышение надежности аппаратуры, уменьшение размеров конструктивных единиц, минимизацию взаимных наводок, задержек сигналов, уменьшение общей длины соединений и т.п.[1].

Создание и использование интегрируемых программных модулей, дополняющих и расширяющих возможности САПР в аспекте автоматизированного проектирования, отвечающего множеству требований, позволит значительно сократить сроки разработки электронных модулей высокого качества.

В основе методов и алгоритмов, реализуемых программным модулем автоматизированного размещения, должны лежать следующие принципы:

– учет функционально-узловой структуры устройств, требований компактного размещения функциональных узлов (ФУ) и выдерживание заданного порядка следования звеньев обработки сигнала, то есть требуемого пути его прохождения;

– учет возможного наличия в схеме структурно-подобных частей (например, одинаковых каналов обработки сигнала или коммутационных линий), требований симметричности размещения таких частей;

– учет возможного наличия в схеме РЭК с особыми признаками: компонентов-источников электромагнитных помех или повышенного тепловыделения, компонентов, положение которых фиксировано (например, разъемы базовых несущих конструкций).

Важное следствие всех этих принципов и, одновременно, основное требование к методам и алгоритмам можно сформулировать так: положение каждого элемента должно быть обосновано с точки зрения качественного и надежного функционирования устройства, а также подчиняться единой конструктивной стратегии размещения. Кроме того, из всех вышеизложенных условий следует, что задача размещения – обязательно многокритериальная задача.

Несмотря на внушительный массив проведенных исследований, задача размещения РЭК до сих пор не теряет свою актуальность, ввиду увеличения множества конструкционных и физических требований, предъявляемых к аппаратуре. Так как задача размещения относится к классу NP-полных, разработка алгоритма, позволяющего найти оптимальное решение за приемлемое время – весьма затруднительна [2].

При создании средств автоматизации размещения в контексте усложненной задачи необходимо применить системный подход в рассмотрении объекта размещения, включающий анализ всех его составляющих, а также условий задачи; поиск аналогий и синтез новых моделей представления и решения. Так, целесообразно рассмотреть объект проектирования как динамически развивающуюся совокупность сущностей и отношений согласно принципам целостности, иерархичности, структурности, множественности и эмерджентности. Одной из перспективных моделей такого представления является модель мультиагентной системы (МАС), которая также позволяет рассматривать компоненты как интеллектуальные сущности. Это, в свою очередь, заметно расширяет поле допустимых частных процедур размещения и его оптимизационных критериев.

Аксиоматика агентно-ориентированного подхода в задаче размещения

Задача построения мультиагентной модели есть задача декомпозиции объекта и представления его как динамической, самоорганизующейся системы. При этом элементы такой системы являются интеллектуальными сущностями, называемые агентами. Агент в контексте задачи интерпретируется как РЭК, а множество всех агентов (то есть МАС) как множество всех размещаемых РЭК. В контексте программной реализации агент – есть программная сущность, объект, экземпляр соответствующего класса.

Агент характеризуется вектором изменяемых и неизменяемых параметров. Значение изменяемой части вектора в конкретный момент времени – есть состояние агента. Для агента, представляющего РЭК в задаче размещения, определены следующие изменяемые параметры: координаты центра РЭК или его полюса в двумерном евклидовом пространстве:

– угол поворота РЭК относительно его начального положения;

– признак стороны размещения (при двустороннем размещении).

Неизменяемыми параметрами являются:

– габариты элемента;

– параметры, характеризующие РЭК с точки зрения особых признаков (электромагнитных помех, тепла и т.д.).

Таким образом, состояние РЭК в конкретный момент времени определяется его положением в пространстве среды размещения.

В контексте программной реализации, вектор параметров агентов задается параметрами класса, экземплярами которого они являются.

Агент способен выполнять некоторый набор действий. Непосредственно, выполнение действий есть поведение агента. Относительно поведения системы, поведение агента примитивно, но поведение всех агентов определяет поведение системы. В сущности, действие агента есть способ изменения значения изменяемой части вектора параметров агента. Таким образом, в результате поведения агента изменяется его состояние. Для агента, представляющего РЭК в задаче размещения, определены следующие действия:

– перемещение (изменение координат центра или полюса);

– поворот относительно центра или полюса;

– изменение стороны размещения.

Так как все возможные действия агента в данном случае связаны с изменением значений параметров, характеризующих его положение в пространстве, поведение агента заключается только в его перемещении в этом пространстве. В программном контексте действия агентов описываются методами класса, экземплярами которого они являются.

Интеллектуальность агента означает, что перед тем, как выполнить любое действие, агент осуществляет принятие решения, алгоритмически реализуемое либо как обработка ситуации в соответствии с заранее определенным набором правил, либо как интеллектуальное планирование или прогноз с учетом издержек при выполнении того или иного действия. В любом случае, выбор следующего действия происходит в результате анализа собственного состояния и состояния среды, в том числе, состояния связанных агентов и агентов, находящихся поблизости в параметрическом пространстве. Эти данные являются входом алгоритма принятия решений, который формирует управляющие воздействия, побуждающие агента выполнять конкретные действия. Такой автомат будем называть вычислительной частью агента (ВЧА).

В задаче размещения, для агентов, представляющих РЭК, к данным о состоянии среды относятся:

– данные состояний связанных агентов, то есть положений РЭК, подключенных к той же цепи;

– данные состояний ближайших агентов, то есть положений РЭК, находящихся в той же окрестности;

– данные состояний агентов с особыми признаками, то есть положений РЭК-источников электромагнитных помех, тепла и т.д.;

– наличия в окрестности агента особых зон среды, для РЭК – запрещенных зон размещения.

Для представления МАС из таких агентов будем использовать модель «цель-регламент» или ЦР-модель. Согласно ЦР-модели, поведение агентов сводится к их перемещению в параметрическом пространстве (в данном случае – среде размещения) согласно набору правил, называемому регламентом, для достижения некоторой цели или выполнения функции.

Цель агента – обнаружить свое оптимальное состояние, а в контексте задачи размещения, занять оптимальное положение в пространстве. Цель МАС (как системы самоорганизующееся) – обнаружить свое оптимальное состояние, выраженное некоторым отображением оптимальных состояний всех агентов. Таким образом, цель системы – есть композиция целей агентов. Суть композиционного, а не аддитивного характера целевой функции МАС состоит еще и в том, что критерии поиска для агента и для МАС – различны.

Регламент есть список правил, руководствуясь которым агент принимает решение, чтобы совершить то или иное действие. Можно сказать, что поведение агента регулируется регламентом. Регламент построен относительно всех возможных ситуаций, описывающих взаимные состояния агента и среды, принимая эти ситуации за условия и отображая их во множество всех возможных регулирующих воздействий. Из этого следует, что поведение агента детерминировано относительно состояния среды. Здесь и далее в понятие «состояние среды» мы будем включать все данные, перечисленные выше. В программном смысле, рег-

ламент – это набор условных операторов, реализующих выбор операции над параметрами агента в зависимости от значений параметров других объектов системы.

Структурные группы и алгоритмы структурирования МАС

Алгоритмическая и программная реализация размещения на основе ЦР-модели подразумевает перебор больших массивов данных, связанный с проверкой состояний всех агентов перед выполнением каждого шага размещения. Ключевыми методами повышения эффективности являются использование программных средств распараллеливания, а также отсечение избыточных операций перебора.

Для возможности применения этих методов необходимо упорядочить и структурировать систему, используя признаки связности и отношений агентов. При этом структурные единицы и принципы их выделения должны легко интерпретироваться в контексте специфики объекта задачи и реализовывать ее условия.

Можно выделить три вида структурных единиц: функциональные группы, структурно-подобные группы и группы с особыми признаками.

Функциональная группа (ФГ) есть подмножество связанных агентов, оптимальные положения которых находятся в общей локальной окрестности параметрического пространства. В контексте задачи размещения, ФГ интерпретируются как функциональные узлы устройства, каждый из которых выполняет некоторую функцию обработки сигнала, и РЭК которого должны быть размещены максимально компактно. Алгоритм выделения таких групп будем называть алгоритмом разбиения. Применение алгоритма разбиения в составе алгоритмической структуры программного модуля позволит решать задачу размещения в соответствии с принципом учета функционально-узловой структуры устройств. Пример функциональной группы (схема источника питания) показана на рисунке 1.

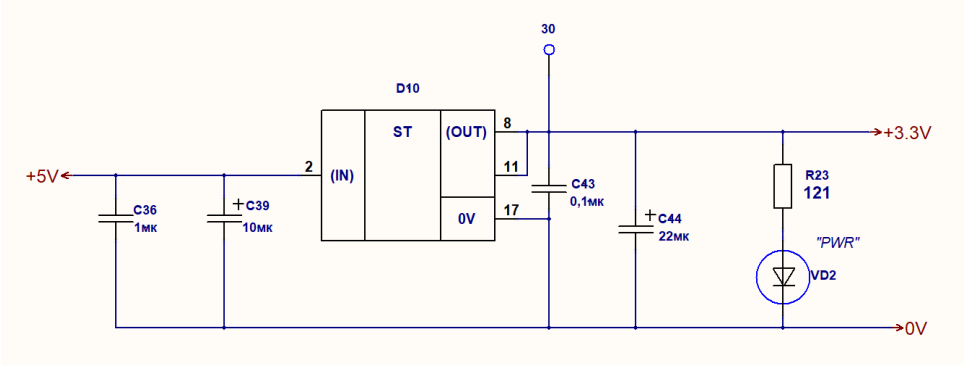


Рис. 1. Пример функциональной группы

Структурно-подобная группа (СПГ) есть множество одинаковых по составу и структуре подмножеств связанных агентов, причем оптимальные положения агентов одного подмножества совпадут с оптимальными положениями другого подмножества при переносе одного на другое. В контексте задачи размещения, СПГ интерпретируется как подобные части схемы (например, каналы обработки сигналов) размещение, которых должно быть максимально симметричным. Алгоритм выделения таких групп будем называть алгоритмом типизации. Применение алгоритма типизации в составе алгоритмической структуры программного модуля позволит решать задачу размещения в соответствии с принципом учета наличия в устройствах подобных узлов.

Пример структурно-подобной группы (фрагмент схемы коммутатора) показан на рисунке 2.

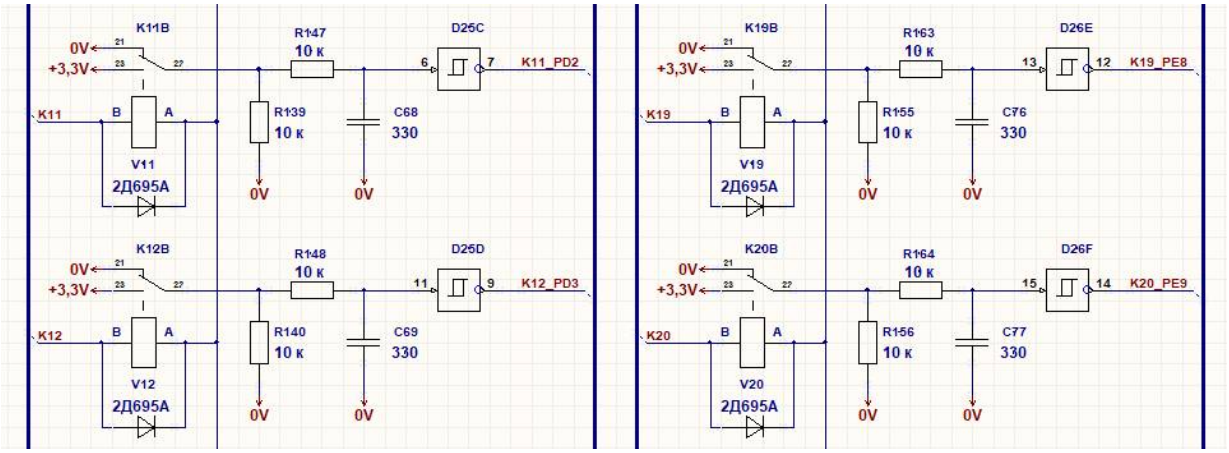


Рис. 2. Пример структурно-подобной группы

Группа с особыми признаками (ГОП) есть подмножество агентов, которые обладают особыми признаками, формирующими особые отношения с агентами вне этого подмножества. В контексте задачи размещения, ГОП интерпретируется как группа РЭК-источников электромагнитных помех, повышенного тепловыделения или зафиксированных РЭК. Оптимальное положение таких компонентов регламентируется специальными правилами. Алгоритм выделения таких групп будем называть алгоритмом назначения. Применение алгоритма назначения в составе алгоритмической структуры программного модуля позволит решать задачу размещения в соответствии с принципом учета наличия в устройствах вышеописанных РЭК.

Алгоритмы разбиения, типизации и назначения являются подготовительными алгоритмами, так как их работа предшествует работе основного алгоритма размещения. Будем называть их алгоритмами структурирования. Необходимость регулирования поведения агентов, принадлежащих определенным структурным группам, должна учитываться при построении регламентов.

Введем особую структурную единицу – кластер – подмножество агентов с общей характеристикой, либо значение некоторой общей количественной характеристики каждого из которых входит в определенный интервал, соответствующий данному кластеру. Причем такая характеристика связанна со структурой МАС. В контексте задачи размещения кластер интерпретируется как подмножество РЭК, количество связей каждого из которых равно определенному значению или входит в определенный интервал. Введения кластеров обосновано следующим:

– необходимость определения формальных центров групп для облегчения реализации их размещения;

– дифференцирование агентов разных кластеров по скорости перемещения (отсутствие такого разделения и перемещение всех агентов с одинаковой скоростью может привести к запутыванию и хаотизации процесса);

– применения механизма распараллеливания (применить механизм к размещению кластеров легче, чем к размещению ФГ и СПГ, так как число кластерных интервалов обычно меньше числа таких групп, а число потоков процессора ограничено);

– возможность задать различные аддитивные критерии оптимизации положения агентов различных кластеров, что позволит оптимизировать процесс размещения путем распараллеливания поисковых процессов.

В алгоритме кластеризации в качестве критерия можно использовать валентности вершин графа ФГ. Агент с самой большой валентностью входит в высший кластер, становится формальным центром группы и самым «медленным» агентом. Принцип дифференциации агентов по скоростям трактуется объектной сущностью агентов. Каждый агент связан с некоторым подмножеством агентов. Перед тем как совершить следующее действие, агент

оценивает состояние связанных с ним агентов, и чем больше таковых, тем больше времени занимает процесс принятия решения. Следовательно, чем мощнее подмножество агентов, связанных с конкретным агентом, тем менее мобилен этот агент, и тем меньше его скорость выполнения операций [3].

Итоговая схема структуры МАС показана на рисунке 3.

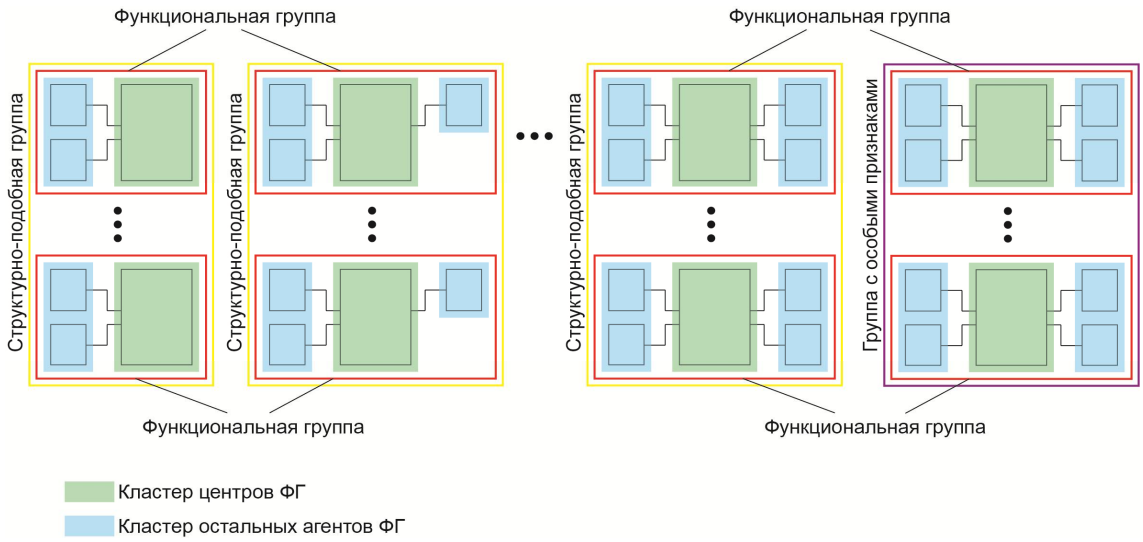


Рис. 3. Структура МАС

После применения алгоритмов структурирования начинается процесс размещения, а в системном смысле – самоорганизация МАС.

Размещение представляет собой детерминированный, квазинепрерывный процесс перемещения агентов в среде согласно регламентам с целью найти свое оптимальное положение в зоне размещения по аддитивному критерию, соответствующему кластеру агента. При этом в ФГ агенты нижних кластеров тяготеют к окрестностям более медленных центров ФГ, которые в свою очередь оптимизируют свое положение относительно друг друга, если считать, что положение центра ФГ задает окрестность размещения всей ФГ. Можно сказать, что на каждом уровне структурной декомпозиции осуществляется своя оптимизация, то есть агенты оптимизируют положение относительно других агентов, а ФГ относительно других ФГ. Кроме этого, в регламентах учтено существование СПГ и ГОП так, что агенты в процессе принятия решений учитывают влияние агентов с особыми признаками, а также свою принадлежность к СПГ.

Алгоритмическая структура и компоненты интегрируемого модуля

Задача интегрируемого программного модуля, как инструментального средства автоматизированного проектирования, состоит в размещении РЭК на печатной плате. Опишем ключевые особенности модуля и представим его обобщённую алгоритмическую структуру.

Для работы модуля необходимы следующие исходные данные:

– список РЭК;
– список электрических цепей;
– габариты РЭК;
– таблица координат выводов РЭК;
– таблица принадлежности контактов РЭК электрическим цепям;
– список РЭК со специальными признаками;
– конфигурация печатной платы и запрещенных зон размещения.

Практически все разделы исходных данных для модуля можно импортировать из САПР. Целесообразно обеспечить модуль собственной базой данных для хранения посадочных мест РЭК. Ее наличие значительно упростит извлечение конфигураций компонентов и их электрических графов.

Для представления информации в виде структур, с которыми работают алгоритмы структурирования и размещения (например гиперграф электрической схемы), данные обрабатываются в предпроцессоре модуля.

Весьма проблематичным является извлечение из САПР данных о РЭК с особыми признаками, так как принципиальная схема не несет в себе такой информации. Для назначения подобных РЭК и, следовательно, для формирования ГОП, целесообразно использовать специальный интерактивный компонент. Таким образом, реализуется процедура назначения. Результаты процедуры назначения влияют на построение аддитивного критерия поиска. Например, если не будут назначены РЭК-источники электромагнитных помех, то в процессе размещения наличие агентов с таким особым признаком учитываться не будет.

Для назначения РЭК с фиксированным положением интерактивный компонент должен быть связан с графической средой размещения модуля.

Контуры печатной платы и запрещенных зон можно определить также с помощью интерактивного компонента (средствами графического редактора), либо импортировать из САПР в стандартном формате передачи чертежей (.dxf, .dwg).

После передачи исходных данных происходит определение параметров агентов и, следовательно, их начальных состояний. Агенты с нефиксированным положением выставляются в среде размещения вне плоскости печатной платы (в процессе размещения агенты будут двигаться в сторону этой плоскости), на верхней стороне платы, с нулевым углом поворота.

Блок построения регламента использует данные, полученные в результате структурирования системы и назначения РЭК с особыми признаками. Он также обеспечен интерактивным компонентом для возможного внесения пользователем заключительных настроек, например, установки оптимизационных критериев, шага дискретной сетки размещения и так далее.

После работы алгоритма укладки, который производит уплотнение размещения и тем самым реализует требование его компактности и минимума площади печатной платы, занимаемой РЭК, происходит вывод полученного плана в графической среде модуля, где он становится доступным для визуального контроля.

Так как алгоритм решения усложненной задачи может достичь лишь локального оптимума целевой функции, модуль представляет только возможные варианты (планы) размещения. Пользователь может корректировать промежуточный план размещения, представляемый в графической среде, и тем самым влиять на процесс работы алгоритма. Так, например, можно фиксировать положение агентов-центов ФГ, в результате чего они станут определяться алгоритмом как элементы с особым признаком, и по отношению к ним начнут действовать специальные правила общего регламента агентов. Таким образом, произойдет «горячая» перестройка регламента, следовательно, самого процесса размещения. Возможность корректировки позволяет с большей степенью контролировать работу алгоритма с учетом усложненных требований.

Обобщенная структура модуля показана на рисунке 4.

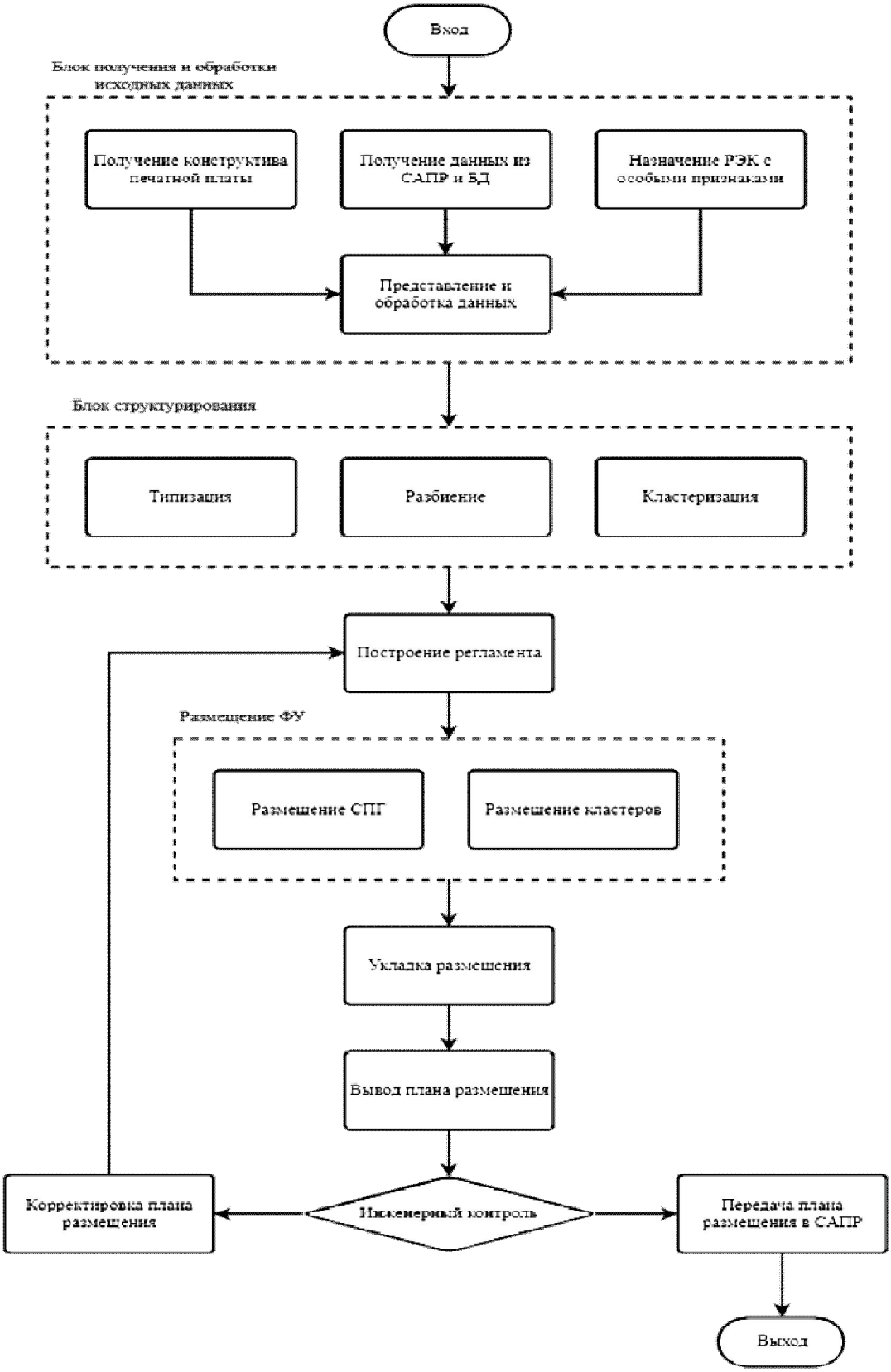


Рис. 4. Структура модуля размещения

Заключение

Создание программного модуля размещения как инструментального средства автоматизированного проектирования топологии печатных плат является актуальной задачей в сфере разработки САПР для использования в наукоемкой промышленности. В условиях ускоряющихся темпов разработки и производства – с одной стороны, и функционального, производительного усложнения устройств – с другой, рассматриваемое программное средство должно совмещать в себе эффективность выполнения задачи и соответствие результатов предъявляемым требованиям. В такой ситуации, использование методов, основанных только лишь на теоретико-множественном и графовом представлениях объекта и решающих задачу с учетом ограниченного числа критериев, уступает применению новых методов, построенных на аналогиях поведения природных и социальных систем. Благодаря расширенному представлению объекта задачи как динамически-развивающейся системы такой подход позволяет модифицировать и сочетать друг с другом известные дискретные модели и синтезировать комплексные алгоритмы многокритериальной оптимизации.

Решение задач проектирования электронных средств с помощью мультиагентных методов и моделей – новейшее направление в исследованиях по данной научно-технической проблематике. Так как теория мультиагентных систем находится на раннем этапе своего развития, который характеризуется существованием множества формализмов – отдельных для каждого применения и отсутствием единого подхода, разработка теоретического и математического базиса для решения конкретной задачи должна начинаться с создания аксиоматики и опорной модели для их последующего развития и, в итоге, вывода законченных алгоритмов.

Работа выполнена при финансовой поддержке гранта Президента РФ МД-3201.2022.1.6.

Библиографический список

1. Курейчик В. М. Математическое обеспечение конструкторского и технологического проектирования с применением САПР: Учебник для вузов. – М.: Радио и связь, 1990.– 352с.: ил.

2. Перепелкин Д.А., Ликучев В.Ю. О многоагентном подходе к решению задач размещения радиоэлектронных компонентов. Информационные технологии, межвузовский сборник научных трудов. – Рязань: ИП Коняхин А.В. (Book Jet), 2021. – 207 с.

3. Перепелкин Д.А., Ликучев В.Ю. Мультиагентный подход автоматизированного проектирования модулей радиоэлектронных устройств на основе анализа инженерных стратегий. V Международный научно-технический форум СТНО-2022. Сборник трудов. Том 3.

УДК 004.72; ГРНТИ 50.41.23

ОПТИМИЗАЦИЯ ГИПЕРПАРАМЕТРОВ РЕКУРРЕНТНОЙ НЕЙРОННОЙ СЕТИ НА ОСНОВЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ ЗАДАЧИ МНОГОПУТЕВОЙ МАРШРУТИЗАЦИИ В ПРОГРАММНО-КОНФИГУРИРУЕМЫХ СЕТЯХ

Д.А. Перепелкин, В.Т. Нгуен

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, nguyenvantinrsreu@gmail.com

Аннотация. В данной работе исследована задача многопутевой маршрутизации в программно-конфигурируемых сетях с использованием рекуррентной нейронной сети, гиперпараметры которой выбираются с помощью генетического алгоритма.

Ключевые слова: программно-конфигурируемые сети, многопутевая маршрутизация, рекуррентная нейронная сеть, генетический алгоритм.

HYPERPARAMETERS OPTIMIZATION OF RECURRENT NEURAL NETWORK BASED ON GENETIC ALGORITHM FOR THE PROBLEM OF MULTIPATH ROUTING IN SOFTWARE DEFINED NETWORKS

D.A. Perepelkin, V.T. Nguyen

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, nguyenvantinrsreu@gmail.com

The summary. In this paper, we study the problem of multipath routing in software-defined networks using recurrent neural network whose hyperparameters are selected using genetic algorithm.

Keywords: software defined networks, multipath routing, recurrent neural network, genetic algorithm.

В последние годы развитие Интернет-технологий приводит к широкому внедрению новых программных приложений, что значительно увеличивает нагрузку на каналы связи компьютерных сетей. Интернет-индустрии нужна новая сетевая архитектура для решения существующих сетевых проблем, и эта архитектура должна быть более гибкой и эффективной. Для решение таких проблем была предложена технология программно-конфигурируемых сетей (ПКС) [1]. На основе изучения современной технологии планирования трафика ПКС, сочетающей в себе преимущества глубокого обучения и возможности централизованного управления сетевыми ресурсами ПКС, в данной статье предлагается интеллектуальный метод многопутевой маршрутизации на основе рекуррентной нейронной сети – LSTM. Экспериментальные результаты показывают, что этот метод может использовать характеристики ПКС для маршрутизации сетевого трафика по нескольким путям в соответствии с информацией о текущем состоянии и характеристиками трафика сети. Используя преимущества глубокого обучения, данный метод может найти несколько путей пересылки данных в режиме реального времени для различных потоков в соответствии с характеристиками их трафика и повысить коэффициент использования пропускной способности сетевого канала. Генетический алгоритм (ГА) используется для выбора гиперпараметров модели [2].

Для подтверждения эффективности предложенного метода рассматривается топология ПКС, показанная на рисунке 1. Задача состоит в том, чтобы найти четыре кратчайших маршрута от сервера H1 к серверу H2. Эта топология ПКС описывается матрицей X следующим образом:

$$X = \begin{bmatrix} d_{11} & \cdots & d_{18} \\ \vdots & \ddots & \vdots \\ d_{81} & \cdots & d_{88} \end{bmatrix},$$

где $d_{ij} \in [10, 20, 30, 40, 50, 60, 70, 80, 90]$ – задержка канала (i, j) (несуществующие каналы будут представлены как 0).

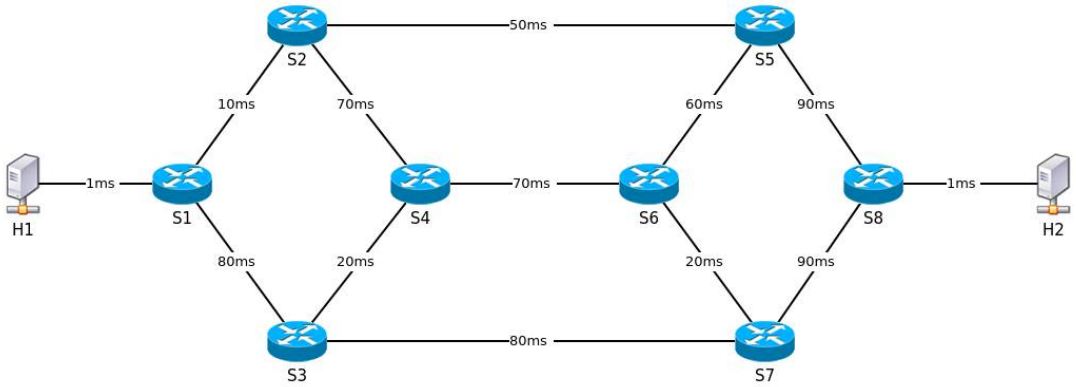


Рис. 1. Рассматриваемая топология ПКС

Для решения поставленной задачи строится модель нейронной сети, как показано на рисунке 2, в которой входными данными (Input) является векторизованная X-матрица. Два слоя рекуррентных нейронных сетей используются для извлечения связи между кратчайшими путями. На выходе модели используется слой полносвязных нейронных сетей, соответствующих каждому пути через метод двоичного кодирования.

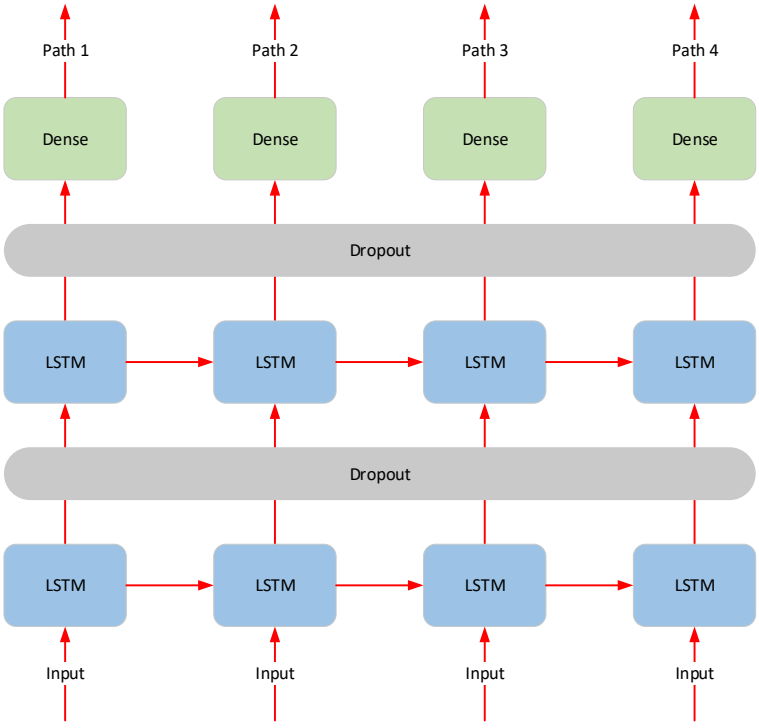


Рис. 2. Модель нейронной сети

Для обучения модели используется набор данных, состоящий из 50000 образцов, из которых 5000 образцов используются для выбора гиперпараметров в генетическом алгоритме, еще 5000 образцов используются в качестве тестового набора, а остальные 40000 образцов используются для обучения модели. Входные данные были нормализованы с использованием метода нормализации StandardScaler. Чтобы избежать переобучения, в структуру модели добавлены два слоя Dropout. Модель обучается с помощью алгоритма оптимизации Adam. Генетический алгоритм используется для оптимизации гиперпараметров. Рассмотренные гиперпараметры модели приведены в таблице 1.

Таблица 1. Рассмотренные гиперпараметры модели

Гиперпараметры	Множество значений
Количество нейронов в первом слое LSTM	[25, 30, 35, 40, 45, 50]
Dropout 1	[0.1, 0.2]
Количество нейронов в втором слое LSTM	[25, 30, 35, 40, 45, 50]
Dropout 2	[0.1, 0.2]
Скорость обучения (learning_rate)	[0.1, 0.05, 0.01, 0.005, 0.001, 0.0005, 0.0001]
beta_1	[0.7, 0.8, 0.9, 0.99, 0.999]
beta_2	[0.9, 0.99, 0.999, 0.9999, 0.99999]
epsilon	[1e-05, 1e-06, 1e-07, 1e-08, 1e-09]
батч (batch size)	[16, 32, 64, 128, 256]

Используемые параметры ГА: размер популяции $N = 10$, количество итераций $Max = 10$, вероятность скрещивания $P_c = 0.7$ и вероятность мутации $P_m = 0.1$. Результаты эксперимента показаны на рисунке 3.

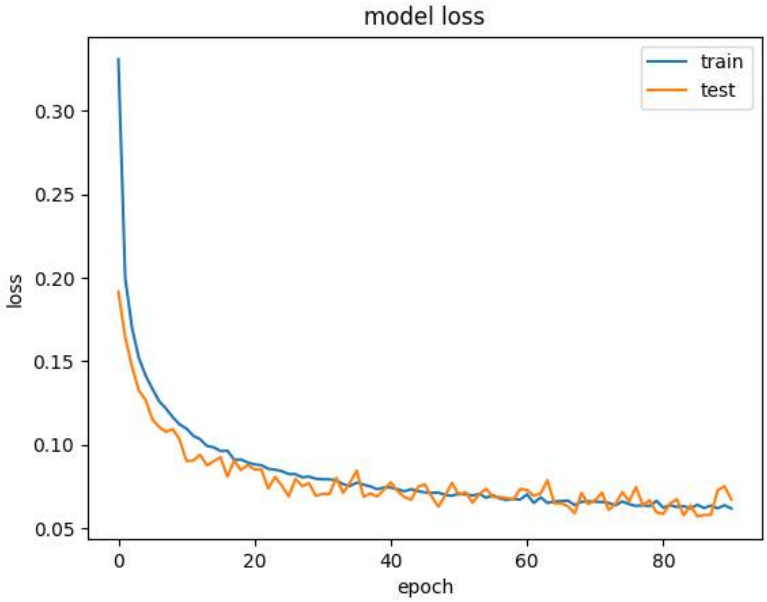


Рис. 3. График значений функции потерь после каждой эпохи

Набор найденных гиперпараметров является: [40, 0.1, 50, 0.1, 0.01, 0.7, 0.9999, 1e-05, 64]. Точность модели нейронной сети после 90 эпох (с применением обратных вызовов callbacks) составляет 89.64% на наборе обучающих данных и 86.06% на наборе тестовых данных.

Исследование выполнено при поддержке гранта Российского научного фонда (РНФ) и Правительства Рязанской области, проект № 22-21-20093.

Библиографический список

1. Mckeown N, Anderson T, Balakrishnan H, et al. OpenFlow: Enabling innovation in campus networks[J]. ACM SIGCOMM Computer Communication Review, 2008, 38(2):69-74.

2. Nurshazlyn Mohd Aszemi, P.D.D Dominic. Hyperparameter Optimization in Convolutional Neural Network using Genetic Algorithms. (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 10, No. 6, 2019.

УДК 004.032.26; ГРНТИ 28.23.37

РАЗРАБОТКА РЕКОМЕНДАТЕЛЬНОЙ СИСТЕМЫ ВЫБОРА НАПРАВЛЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ АБИТУРИЕНТАМИ

М.А. Иванчикова, Д.С. Венчикова

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, tyrbinad@gmail.com

Аннотация. В работе рассматриваются архитектура нейронной сети и методы машинного обучения для разработки рекомендательной системы выбора направления высшего образования абитуриентами. Приводятся их основные особенности, достоинства и недостатки, а также принципы работы.

Ключевые слова: рекомендательная система, высшее образование, машинное обучение, нейронные сети, абитуриент.

DEVELOPMENT OF A RECOMMENDATION SYSTEM FOR CHOOSING THE DIRECTION OF HIGHER EDUCATION BY APPLICANTS

M.A. Ivanchikova, D.S. Venchikova

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, tyrbinad@gmail.com

The summary. The paper discusses the architecture of a neural network and machine learning methods for developing a recommendation system for choosing the direction of higher education by applicants. Their main features, advantages and disadvantages, as well as principles of operation are given.

Keywords: recommendation system, higher education, machine learning, neural networks, entrant.

Привлечение школьников к поступлению в высшее учебное заведение, увеличение количества абитуриентов и прогнозирование их потока является одной из актуальных задач высшего образования. Разработка рекомендательной системы позволит повысить качество приемной кампании и поможет выявить сильные и слабые стороны в принятии управленческих решений относительно контрольных цифр приема на отдельные направления.

Целью работы является разработка интеллектуальной рекомендательной системы, основанной на алгоритмах и методах машинного обучения, предназначенной для решения задач высшего образования. На первом этапе исследования был проведен сбор исходной информации, анализ и предобработка данных, сформирован датасет и выборки [1]. Следующим этапом является обучение модуля системы, производящего классификацию данных. В докладе рассматривается выбор архитектуры нейронной сети и метода машинного обучения и подбор гиперпараметров, используемые для разработки рекомендательной системы выбора направления высшего образования абитуриентами.

Для выбора оптимальной архитектуры проводился сравнительный анализ сетей. В итоге была выбрана рекурентная нейронная сеть (RNN), так как задачей является создание рекомендательной системы, основанной на прогнозировании одномерного временного ряда с небольшим количеством входных данных.

Рекуррентные нейронные сети (RNN) специализируются на обработке последовательностей. Их основным достоинством является возможность обрабатывать последовательности с вариативными длинами как для входа, так и для вывода. В отличие от прямой нейронной сети RNN является вариантом рекурсивной нейронной сети, в которой связи между нейронами реализованы направленными циклами. Поэтому выходная информация зависит не только от текущего входа, но также от состояний нейрона на предыдущем шаге.

В традиционной нейронной сети входные и выходные данные независимы друг от друга, тогда как выходные данные в RNN зависят от предыдущих элементов в последовательности. Рекуррентные сети также совместно используют параметры на каждом уровне сети. В сетях с прямой связью для каждого узла существуют разные веса, а RNN разделяет

одинаковые веса на каждом уровне сети и во время градиентного спуска веса и базис корректируются индивидуально, чтобы уменьшить потери.

Для подбора гиперпараметров обучения нейронной сети была использована библиотека Keras Tuner. Keras Tuner – это библиотека, которая поможет подобрать оптимальный набор гиперпараметров для программы, разработанной с помощью библиотеки TensorFlow.

Гиперпараметры – это переменные, которые управляют процессом обучения и топологией модели машинного обучения. Эти переменные остаются постоянными в процессе обучения и напрямую влияют на производительность вашей программы машинного обучения. Гиперпараметры бывают двух типов:

1. Гиперпараметры модели, влияющие на выбор модели, такие как количество и ширина скрытых слоев.
2. Гиперпараметры алгоритма, которые влияют на скорость и качество алгоритма обучения, такие как скорость обучения для стохастического градиентного спуска (SGD) и количество ближайших соседей для классификатора ближайших соседей (KNN).

Для работы с библиотекой для начала необходимо ее скачать и загрузить данные.

Далее определяем модель. В процессе построения модели для гипернастройки, также определяем пространство поиска гиперпараметров в дополнение к архитектуре модели. Модель, которую устанавливается для гипернастройки, называется гипермоделью.

Гипермодель можно определить с помощью двух подходов:

- Используя функцию построителя модели;
- Путем создания подкласса класса HyperModel API Keras Tuner.

Функция построителя модели возвращает скомпилированную модель и использует гиперпараметры, которые вы определяете встроено, для гипернастройки модели.

Keras Tuner имеет четыре доступных тюнера — Hyperband, BayesianOptimization, RandomSearch, Sklearn. Выберем Hyperband, относящийся к Байесовской реализации, и в основе которого лежит алгоритм деления пополам с различными бюджетами. Выбран был именно Hyperband, так как с помощью данного метода не обучается полностью каждый вариант сети, а обучение проходит частично на малом количестве эпох, и затем отсеивается какое-то количество более худших вариантов, обучая дальше лишь лучшие, и так отбор происходит какое-то количество раз (эпох). Подбираем максимальное количество эпох для обучения. Для исходного датасета взяла 10 (Листинг 1).

Листинг 1 – Параметры Hyperband

```
Tuner = kt.Hyperband(model_builder,
objective='val_accuracy',
max_epochs=10,
factor=3,
directory='my_dir',
project_name='intro_to_kt')
```

Создаем обратный вызов, чтобы остановить обучение досрочно после достижения определенного значения потери проверки (Листинг 2).

Листинг 2 – Создание обратного вызова

```
stop_early = tf.keras.callbacks.EarlyStopping(monitor='val_loss', patience=5)
```

Далее запускаем поиск оптимальных гиперпараметров. После этого нужно найти оптимальное количество эпох для обучения модели с гимперпараметрами, полученными в процессе исследования.

В процессе изучения методов машинного обучения для решения поставленной задачи были выбраны наиболее известные алгоритмы классификации: метод опорных векторов (Support Vector Machines – SVM), метод главных компонент (Principal Components Analysis – PCA), AdaBoost. Данные алгоритмы использовались на практике для реализации рекомендательной системы выбора направления высшего образования абитуриентами. В процессе обу-

чения входные данные были еще раз пересмотрены с учетом их влияния на время и качество обучения системы.

В результате сравнительного анализа было выявлено, что самым эффективным алгоритмом машинного обучения для разработки рекомендательной системы выбора направления высшего образования является алгоритм AdaBoost. Данный алгоритм лучше всего работает со слабыми обучающими алгоритмами, поэтому такие модели достигают точности гораздо выше случайной при решении задачи классификации. Слабый обучающий алгоритм – это классификатор или алгоритм предсказания, который работает относительно плохо в плане точности. Чаще всего в комплексе с AdaBoost применяются одноуровневые деревья решений. Слабые классификаторы легко вычисляются, поэтому можно объединять несколько сущностей алгоритма, чтобы создать более сильный классификатор с помощью бустинга. Для объединения результатов алгоритм присваивает вес каждому классификатору в зависимости от полученного ответа на предыдущем этапе. Дополнительным преимуществом данного алгоритма является то, что его легко, быстро и просто запрограммировать. Кроме того, он достаточно гибкий, чтобы комбинировать его с любым алгоритмом машинного обучения без настройки параметров и его можно использовать с числовыми или текстовыми данными. При практической реализации данного алгоритма была достигнута максимальная точность обучения на тестовом наборе данных – 75%.

Результатами проделанной работы является процесс обучения обучающего набора данных, результат которого 0,73%. Точность обучения валидационного набора данных – 67%. Начата разработка разработка веб-приложения для удобной и интерактивной работы с обученной нейронной сетью и практического применения. Для реализации приложения был выбран фреймворк streamlit. Среди различных библиотек и фреймворков выбран был именно он, потому что streamlit предназначенн для исследования данных и для простого развертывания моделей и визуализаций с использованием Python. Помимо простоты работы с ним, самое главное, данный фреймворк позволяет подключить обученную нейронную сеть.

С помощью веб-приложения абитуриент или работник приемной комиссии сможет загрузить необходимые данные. Внутренние алгоритмы приложения их обработают до данных типа, принимаемых на вход обученной модели (так, например, значения NaN образуются в 0), далее с помощью обученной нейронной сети на выходе пользователь получит рекомендацию по направлению обучения на факультете вычислительной техники.

Также с помощью данной нейронной сети и приложению можно будет прогнозировать поток абитуриентов на следующий год.

Работа выполнена при финансовой поддержке стипендии Президента РФ СП-2459.2021.5

Библиографический список

1. Иванчикова М.А., Турбина Д.С. Предобработка данных для прогнозирования потока абитуриентов по направлениям подготовки высшего образования / Новый информационные технологии в научных исследованиях (НИТ-2020): материалы XXV Всероссийской научно-технической конференции студентов, молодых ученых и специалистов. Рязань: РГРТУ, 2020.
2. Введение в Keras Tuner [Электронный ресурс]: URL: https://www.tensorflow.org/tutorials/keras/keras_tuner
3. Реккурентные нейронные сети (RNN) с Keras [Электронный ресурс]: URL: https://www.tensorflow.org/guide/keras/rnn

УДК 004.032.26; ГРНТИ 28.23.37

ПРИМЕНЕНИЕ НЕЙРОННОЙ СЕТИ ДОЛГОЙ КРАТКОСРОЧНОЙ ПАМЯТИ ДЛЯ ГЕНЕРАЦИИ HTML КОДА САЙТА

М.А. Иванчикова, Д.В. Шаронов
Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, capybara20.07@mail.ru

Аннотация. В работе рассматривается нейронная сеть долгой краткосрочной памяти. Приводится её внутренняя структура, а также процесс её обучения и применения для генерации HTML-разметки сайта по изображению его макета.
Ключевые слова: нейронная сеть долгой краткосрочной памяти, LSTM, HTML-разметка, прямое кодирование, One-Hot Encoding.

APPLICATION OF LONG SHORT-TERM MEMORY NEURAL NETWORK FOR HTML SITE'S CODE GENERATION

M.A. Ivanchikova, D.V. Sharonov
Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, capybara20.07@mail.ru

The summary. The paper considers a neural network of long-term short-term memory. Given its internal structure, and the process of learning and using to generate the HTML markup of the site layout image.
Keywords: neural network of long short term memory (LSTM), HTML markup, One-Hot Encoding.

В настоящее время стремительно развивается применение нейронных сетей для решения задач в разных сферах человеческой деятельности: распознавание образов, принятие решений, прогнозирование, оптимизация и т.д. Одной из перспективных идей является генерация кода с помощью искуственного интеллекта. Для ее реализации целесообразно применять методы машинного обучения и синтезированные наборы данных, которые позволят автоматизировать и значительно ускорить процесс разработки. Чаще всего для решения таких задач применяют рекуррентные нейронные сети, но в случае генерации кода они ряд недостатков. При каждой передаче со слоя на слой контекст постепенно размывается, т.к. в него постоянно что-то записывается, заменяя при этом старую информацию. Вектор состояния записывает всё больше данных в неизменяемое количество выделенных под него нулей и единиц. При постоянной перезаписи информация спрессовывается всё плотнее, что ведёт к потерям. При этом система не различает важные данные контекста от неважных, запоминая всё подряд [1]. Решением данных проблем стало использование нейронной сети долгой краткосрочной памяти LSTM (Long Short Term Memory). Данная сеть самостоятельно определяет, какую информацию следует записать, а какую удалить на каждом шаге работы.

Схема LSTM сети представлена на рисунке 1.

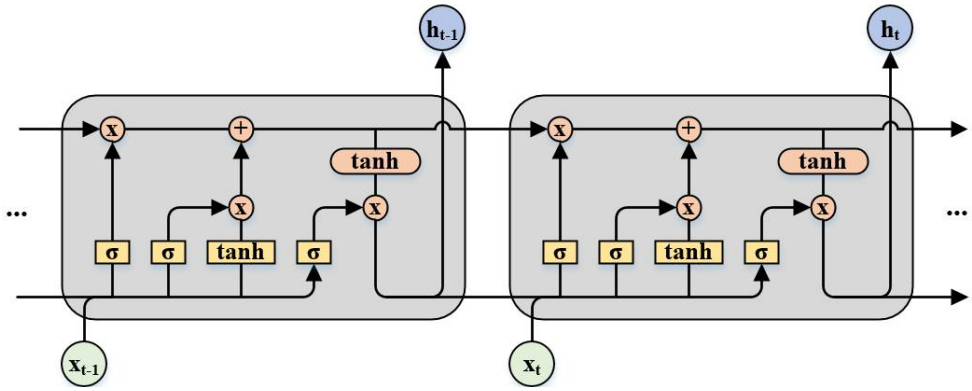


Рис. 1. Схема нейронной сети долгой краткосрочной памяти

Вход x_t (новое входное слово) и выход h_t (предсказание) каждого модуля нейросети представляют собой векторы. Каждый слой нейронной сети состоит из трёх операций:

1) Умножение входного вектора на соответствующую матрицу весов;

2) Прибавление к произведению сдвига;

3) Поэлементное применение к элементам полученного вектора функции активации нейронов (функции сигмоиды или гиперболического тангенса).

Так же каждый модуль содержит ряд посимвольных операций:

1) Определение для каждого элемента вектора значения от 0 до 1:

$$\sigma(x) = \frac{1}{1 + e^{-x}},$$

2) Определение для каждого элемента вектора значения от -1 до 1:

$$\tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1},$$

3) поэлементное умножение векторов;

4) поэлементное сложение векторов.

Отличительной особенностью LSTM сети является проходящий из слоя в слой вектор памяти, часть которого становится предсказанием h_t.

Работа модуля нейронной сети начинается с оценки, какой контекст ей больше не нужен. Эта оценка реализуется сигмоидальным слоем, называемым «слоем фильтра забывания» (Forget Gate Layer). Он получает предсказание предыдущего слоя h_{t-1} и текущее входное слово x_t и возвращает оценку (число от 0 до 1) для каждого числа вектора памяти C_{t-1}, где 1 – «полностью сохранить», а 0 – «полностью выбросить». Оценка вычисляется по формуле:

$$f_t = \sigma\big(W_f * [h_{t-1}, x_t] + b_f\big),$$

где h_{t-1} – предсказание прошлого слоя (вектор);

x_t – новое входное слово (вектор);

W_f – веса забывания (матрица весов);

b_f – сдвиг (вектор);

f_t – оценочный вектор забывания (вектор).

Далее оценивается, какую информацию необходимо запомнить. Эта оценка реализуется двумя независимыми слоями. Сначала сигмоидальный слой, называемый «слоем входного фильтра» (Input Layer Gate) определяет, какие значения нужно обновить. После чего

tanh-слой строит вектор новых значений-кандидатов C_t', которые можно добавить в вектор памяти. После определения того, какую информацию необходимо забыть, а какую записать, производится изменение вектора памяти.

Старый вектор памяти C_{t-1} заменяется на новый C_t. Новое значение вычисляется по формуле:

$$C_t = f_t * C_{t-1} + i_t * C_t',$$

где C_{t-1} – вектор памяти прошлого слоя (вектор);
f_t – оценочный вектор забывания (вектор);
C_t' – вектор новых значений (вектор);
i_t – оценочный вектор записи (вектор);
C_t – вектор памяти (вектор).

В конце определяется, какая часть вектора памяти станет предсказанием. Сначала сигмоидальный слой, называемый «слоем выходного фильтра» (Output Layer Gate) определяет, какие элементы вектора состояния нужно вывести. Затем значения вектора состояния проходят через tanh-слой и перемножаются с оценочным вектором сигмоидального слоя, что позволяет выводить только требуемую информацию.

Слой выходного фильтра возвращает оценочный вектор, который вычисляется по формуле:

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o),$$

где h_{t-1} – предсказание прошлого слоя (вектор);
x_t – новое входное слово (вектор);
W_o – веса выхода (матрица весов);
b_c – сдвиг (вектор);
o_t – оценочный вектор выхода (вектор).

Предсказание вычисляется по формуле:

$$h_t = o_t * \tanh(C_t),$$

где o_t – оценочный вектор выхода (вектор);
C_t – вектор памяти (вектор);
h_t – предсказание (вектор).

При обучении нейронной сети на вход подается несколько идентичных изображений макетов сайтов. На них система учится последовательно предугадывать все используемые теги разметки. При предсказании очередного тега, алгоритму передаётся изображение нового макета, а также правильные теги HTML-разметки сайта с предыдущего шага.

Нейронная сеть создаёт признаки данных. Эти признаки формируют связи между входными и выходными значениями. Для определения содержания изображения сеть создает представления данных, на основании которых генерируется HTML синтаксис предсказания. Так у алгоритма появляются необходимые знания для предсказания следующего тега.

Изображение макета сайта представляется в виде списков пиксельных значений от 0 до 255 в RGB-спектре. Для представления HTML-разметки в понятном алгоритму виде, применяется прямое кодирование (One-Hot Encoding) [2]. Рассмотрим на примере HTML-разметки вида: <start><html>Hello</html><end>. Пример обучения нейронной сети для генерации HTML-разметки представлен на рисунке 2.

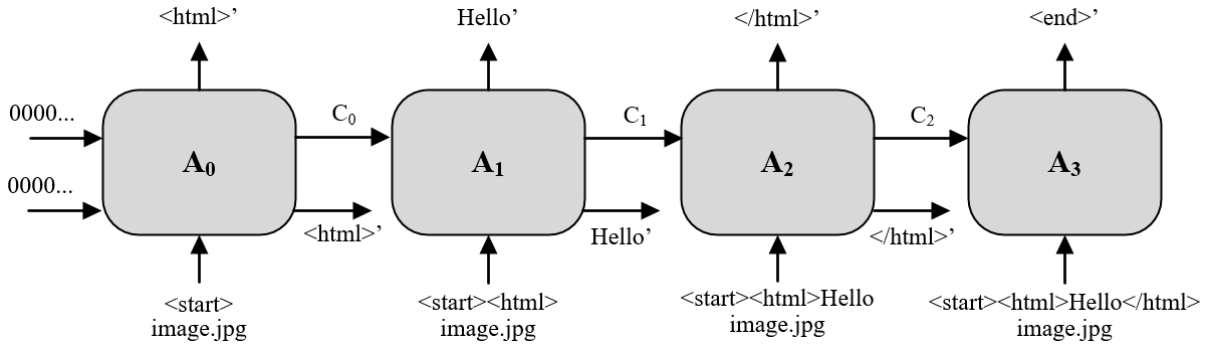


Рис. 2. Обучение нейронной сети на HTML-разметке примера

Текст генерируется пошагово, сопровождается одинаковыми картинками. Вместо передачи алгоритму правильных HTML тегов, она получает уже сгенерированную разметку. Предсказание инициируется начальным тегом и заканчивается конечным тегом, либо достижением максимального размера разметки.

Работа выполнена при финансовой поддержке стипендии Президента РФ СП-2459.2021.5

Библиографический список

1. Christopher Olah, Understanding LSTM Networks. [Электронный ресурс]. – 2015. – Режим доступа: https://colah.github.io/posts/2015-08-Understanding-LSTMs/ – Дата доступа: 10.02.2023.

2. Шаронов Д.В. Обучение нейронной сети для генерации сайта / Современные технологии в науке и образовании (СТНО-2022): труды V Международного научно-технического форума "Современные технологии в науке и образовании" СТНО-2022. Рязань: РГРТУ, 2022.

СЕКЦИЯ «ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ЗАЩИТА ИНФОРМАЦИИ»

УДК 004.056; ГРНТИ 81.93.29

БЕЗОПАСНОЕ ПО И РАЗРАБОТКА БЕЗОПАСНОГО ПО

Ю.М. Кузьмин
Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, kuzmin_yurii@mail.ru

Аннотация. В работе рассматриваются вопросы разработки безопасного программного обеспечения. Рассматриваются требования к разработке и обеспечению безопасности ПО, задачи, технологии и инструменты обеспечения безопасности ПО.
Ключевые слова: жизненный цикл ПО (ЖЦПО), безопасность ПО, уязвимость.

SAFE SOFTWARE AND DEVELOPMENT OF SAFE SOFTWARE

Y.M. Kuzmin
Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, kuzmin_yurii@mail.ru

The summary. In work questions of development of the safe software are considered. Requirements to development and safety of software, a tasks, technology and instruments of safety of software are considered.
Keywords: software development life cycle (SDLC), application security (AppSec), vulnerability.

Необходимость обеспечения безопасности программного обеспечения (ПО) продиктована следующими обстоятельствами:

- растущая сложность ПО и информационных систем, его эксплуатирующих, увеличивает возможности появления уязвимостей в разрабатываемом ПО на разных стадиях жизненного цикла (ЖЦПО);
- эксплуатация уязвимостей в небезопасном ПО может приводить к широкому спектру угроз для пользователей такого ПО (получение возможности удаленного управления системами пользователя, утечка информации, несанкционированная модификация или удаление информации и т.п.);
- не позаботившиеся о безопасности своего ПО разработчики рискуют попасть под действие Уголовного Кодекса РФ, если их ПО, например, копирует данные пользователя без его явного разрешения (статья 272 УК «Неправомерный доступ к компьютерной информации», ч.1);
- уже разработаны международные и отечественные руководящие документы и ГОСТы, описывающие действия, реализация которых должна препятствовать появлению в ПО уязвимостей на всех этапах ЖЦПО;
- уже разработаны международные и отечественные руководящие документы и ГОСТы, описывающие действия и инструменты для выявления оставшихся уязвимостей в разработанном ПО;
- выпущены Постановления Правительства РФ, ФСТЭК России и Банка России, прямо требующие использования процессов безопасной разработки при создании прикладного и системного ПО.

Согласно ГОСТ Р 56939-2016 [1] «**Безопасное ПО -** ПО, разработанное с использованием совокупности мер, направленных на предотвращение появления и устранение уязвимостей программы.»

На практике при создании безопасного ПО используются два направления (два набора мероприятий):

1) меропрятия по предотвращению появления уязвимостей ПО;

2) меропрятия по выявлению уязвимостей в разработанном ПО и анализу риска от эксплуатации выявленных уязвимостей.

Рекомендуется сдвигать мероприятия по обеспечению безопасности ПО влево (shift left) в отношении работ ЖЦПО: чем раньше будут выполняться эти мероприятия, тем меньше потом придется тратить усилий и средств на приведение ПО к соответствию требованиям безопасности.

Мероприятия по предотвращению появления уязвимостей ПО

Порядок и содержание мероприятий данного направления регулируется рядом документов.

1. ГОСТ Р 56939-2016 [1].

2. Приказ ФСТЭК России от 25 декабря 2017 г. № 239 [2].

С 1 января 2023 г. вступили в силу требования этого Приказа] о том, что для значимых объектов КИИ (которые либо создаются либо модернизируются) необходимо прикладное ПО разрабатывать и поставлять безопасным способом (*п.29.3 «...Прикладное ПО должно соответствовать требованиям безопасной разработки...»*).

3. Документ NIST SP 800-218 [3].

Этапы и мероприятия по безопасной разработке ПО на всех этапах ЖЦ ПО согласно ГОСТ Р 56939-2016 приведены в таблице 1.

Таблица 1 Этапы и мероприятия по безопасной разработке ПО согласно ГОСТ Р 56939-2016

Этап	Мероприятия на этапе
Этап 1 «Анализ требований к ПО»	1. Определение требований по безопасности, предъявляемые к разрабатываемому ПО (к идентификация и аутентификация, к защите от НСД к информации; к регистрации событий; к контролю точности, полноты и правильности входных данных ПО; к обработке программных ошибок и исключительных ситуаций).
Этап 2 «Проектирование архитектуры ПО»	2. Моделирование угроз безопасности информации с целью выявления потенциальных угроз безопасности информации, которые вследствие эксплуатации уязвимостей могут возникнуть при запуске ПО. Моделирование угроз проводится для выявления угроз безопасности информации, вызванных ошибками в проекте архитектуры ПО (в архитектуре ПО как логической структуре ПО, в которой определены компоненты ПО, их интерфейсы и взаимодействие между ними по данным и по управлению). Так как моделирование угроз выполняется на ранней стадии разработки ПО (до начала написания кода), то результаты этой работы понадобятся для уточнения проекта архитектуры программы без доработки исходного кода программы. 3. Уточнение проекта архитектуры ПО с учетом необходимости устранения выявленных при моделировании угроз безопасности информации и выполнения требований по безопасности ПО, установленных на этапе 1.
Этап 3 «Конструирование и комплексирование ПО»	4. Использование при разработке ПО только идентифицированных инструментальных средств разработки с определенными опциями (настройками). 5. Создание ПО на основе уточненного проекта архитектуры. Разработчик ПО должен привести в документации сведения о прослеживаемости: соответствие в исходном коде ПО для каждой особенности проекта архитектуры ПО и описанию конкретных проектных решений, обеспечивающих выполнение конкретных требований по безопасности, установленных в процессе анализа требований к ПО на этапе 1. 6. Создание (выбор) и использование при разработке ПО определенного (правильного) порядка оформления исходного кода ПО, содержащего перечень правил и рекомендаций, направленных на устранение (недопущение появления) потенциально уязвимых конструкций в исходном коде ПО. 7. Статический анализ исходного кода ПО (и компонентов от сторонних разработчиков) для выявления и устранения потенциально уязвимых конструкций в исходном коде, которые могут привести к наличию уязвимости ПО. 8. Периодическая экспертиза исходного кода ПО (и компонентов от сторонних организаций). Поскольку экспертиза исходного кода ПО является трудоемкой процедурой, ее обычно проводят в отношении отдельных частей исходного кода ПО. Выбор конкретных частей можно производить исходя из степени важности (критичности) той или иной части с точки зрения обеспечения безопасности разрабатываемого ПО.

Продолжение таблицы 1

Этап 4 «Квалификационное тестирование ПО»	9. Тестирование ПО на безопасность (функциональное) для проверки того, насколько выполняются требования безопасности ПО, установленные на этапе 1. 10. Тестирование на проникновение для выявления (подтверждения) уже не только потенциальных уязвимостей ПО. 11. Динамический анализ кода ПО для выявления уязвимостей (в том числе уязвимостей, не обнаруженных при статическом анализе). 12. Фаззинг ПО. Тесты, выполняемые в рамках тестирования на проникновение, динамического анализа кода и фаззинг, должны быть разработаны с учетом: • проекта архитектуры ПО (с учетом использования компонент от сторонних разработчиков); • перечня потенциальных угроз безопасности информации, выявленных в рамках мероприятий на предыдущем этапе.
Этап 5 «Инсталляция и приемка ПО»	13. Обеспечение защиты ПО от угрозы нарушения целостности в процессе передачи ПО пользователю. Обеспечение целостности требуется для обеспечения соответствия экземпляра ПО, переданного разработчиком, и экземпляра ПО, полученного пользователем. Для реализации данной меры может использоваться, например, вычисление и последующую проверку контрольной суммы поставляемого на носителе информации установочного комплекта ПО. 14. Поставка пользователю эксплуатационных документов в объеме, достаточном для правильной настройки и безопасного применения программы. В эксплуатационных документах должны быть определены перечень и эталонные значения конфигурационных параметров программы.
Этап 6 «Поддержка в процессе эксплуатации»	15. Отслеживание и исправление ошибок и уязвимостей ПО при получения от пользователей ПО сообщений об ошибках в ходе эксплуатации и сопровождения ПО. Также разработчик ПО обязан передавать пользователям: - сведения об обнаруженных им и другими пользователями уязвимостях ПО; - рекомендации по устранению обнаруженных уязвимостей, например, за счет обновления ПО. 16. Систематический поиск уязвимостей ПО разработчиком ПО в ходе эксплуатации и сопровождения ПО.
Этап 7 «Менеджмент документации и конфигурации»	17. Уникальная маркировка каждой версии ПО. 18. Использование системы управления конфигурацией ПО, позволяющей однозначно идентифицировать каждый элемент конфигурации. К элементам конфигурации относятся: а) программа (установочный комплект программы); б) программные и эксплуатационные документы; в) исходный код программы; г) программные и загрузочные модули, в том числе модули сторонних разработчиков ПО; д) инструментальные средства, используемые при разработке ПО; е) информация об с обновлениях ПО и устранениях в ПО выявленных уязвимостей ж) перечень выявленных уязвимостей и НДВ ПО. Задача данного этапа - обеспечить контроль целостности элементов конфигурации.
Этап 8 «Менеджмент среды разработки»	19. Защита от НСД к элементам конфигурации техническими средствами и организационными мерами. 20. Резервное копирование (и, если надо, восстановление) элементов конфигурации. 21. Регистрация фактов изменений элементов конфигурации. Обычно фиксируют: кто вносит изменения, какой элемент конфигурации изменяется, когда внесены изменения.
Этап 9 «Менеджмент персонала»	22. Периодическое обучение сотрудников для поддержания и улучшения компетентности сотрудников организации-разработчика ПО в области разработки безопасного ПО (моделирование угроз безопасности информации, экспертиза исходного кода программы, тестирование на проникновение, статический анализ исходного кода программы, динамический анализ и фаззинг кода программы). 23. Периодический анализ программы обучения сотрудников.

Чем отличается Application Security, SSDLC и DevSecOps?

SDLC (software development life cycle) - это ЖЦ разработки ПО. SDLC включает шаги: формирование требований, проектирование, разработка и тестирование, ввод в эксплуата-

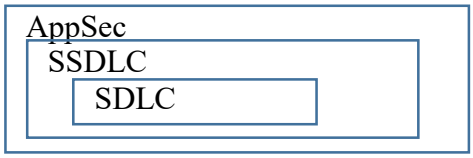
цию, сопровождение, внесение изменений или вывод из эксплуатации (в случает завершения ЖЦ).

SSDLC - это ЖЦ безопасной разработки ПО, где на каждом этапе ЖЦ ПО учитываются требования безопасности и принимаются определенные меры безопасности.

Application Seecurity (AppSec) - это раздел кибербезопасности, в котором объектом защиты является программное обеспечение и его потенциальные уязвимости. Основное внимание в AppSec уделяется предотвращению появления уязвимостей в ПО, а не их обнаружению в уже готовом продукте. AppSec - это значительно большее, чем набор инструментов и методологий (методики). Это в значительной степени осознанный выбор выработанная в организации культура разработки безопасности ПО. Поэтому организации должно быть важным о развитии сотрудников (в вопросах обеспечении безопасности). Это развитие касается всех участников процесса разработки ПО - от архитекторов до тестировщиков, разработчиков и сотрудников службы безопасности информации. Для создания безопасного ПО необходимо, чтобы компетенции ИБ и понимание угроз безопасности присутствовали у всех участников процесса производства ПО.

DevSecOps – это конвейер безопасной разработка приложений. В отличие от SSDLC он не включает разработку требований к безопасности ПО. DevSecOps является частью SSDLC для конвейера DevOps, в который добавлены технологии информационной безопасности. Причем они (технологии) присутствуют на всех этапах - от проектирования до выпуска продукта - вставляются автоматические проверки для выявления небезопасного исполнения языковых конструкций.

AppSec включает SSDLC. SSDLC включает SDLC. Графически это обычно изабражают так:



Меропрятия по выявлению уязвимостей в разработанном ПО

Порядок и содержание мероприятий по данному направлению регулируется следующими документами.

ГОСТ Р 15408-3-2013 [4].

ГОСТ Р ИСО/МЭК 27034 [5] (части 1, 2, 3, 5, 6, 7):

часть 1 (27034-1-2014) Обзор и общие положения;

часть 2 (27034-2-2021) Нормативная структура организации;

часть 3 (27034-3-2021) Процесс менеджмента безопасности приложений;

часть 5 (27034-5-2020) Структуры данных протоколов и мер обеспечения безопасности приложений;

часть 6 (27034-6-2021) Практические примеры;

часть 7 (27034-7-2020) Основы прогнозирования доверия.

ГОСТ Р ИСО-МЭК 18045—2013 [6]. Описывает оценку уязвимостей: устанавливает конкретные шаги оценивания, связанные с действием "Тестирование проникновения" (в подпунктах 14.2.1.6, 14.2.2.7, 14.2.3.7, 14.2.4.7). Оценщик должен разработать тесты проникновения, основываясь на проведенном независимом поиске потенциальных уязвимостей.

ГОСТ Р 58142-2018 [7]. В данном документе определяется порядок проведения тестирования (в том числе тестирование на проникновение) на уязвимости ПО.

Детализация уязвимостей (угроз) приведена в ГОСТ Р 58412.

ГОСТ Р 58412-2019 [8]. В Приложении к этому документу приведена таблица соответствия угроз безопасности информации и мер обеспечения безопасности.

ГОСТ Р 58143-2018 [9].
ГОСТ Р 56545-2015 [10].
ГОСТ Р 56546-2015 [11].
ГОСТ Р 51188-98 [12].

Методический документ ФСТЭК России «**Методика тестирования обновлений** безопасности программных, программно-аппаратных средств» [13].

Приказ ФСТЭК России от 02.06.2020 г. №76 [14]. В этом же приказе предъявляются и требования и по безопасной разработке ПО.

Методический документ ФСТЭК России «**Методика выявления уязвимостей и недекларированных возможностей в ПО»** [15]**.**

Выявление уязвимостей по Методике выявления уязвимостей и НДВ в ПО (далее - Методика)

01.02.2019 ФСТЭК России утвердил «**Методику выявления уязвимостей и недекларированных возможностей в ПО**». 25.12.2020 г. утверждена новая редакция Методики (действует с 1.04.2021).

Содержание Методики (в основном - первой ее редакции от 2019 г.) рассмотрено в работах [16 - 19].

В Приложении к методике описаны методы и средства выявления уязвимостей, методика расчета потенциала нападения, классификации уязвимостей и НДВ. Методы выявления уязвимостей делятся на целевые и вспомогательные. Целевые непосредственно позволяют выявить уязвимость или недекларированную возможность. Вспомогательные улучшают эффективность целевого метода или предоставляют ему некие исходные данные.

При анализе ПО на наличие уязвимостей и НДВ для признания ПО безопасным должно быть подтверждено отсутствия в нем:

а) потенциально опасных функциональных возможностей;

б) опасных функциональных объектов;

в) уязвимостей конфигурации;

г) архитектурных уязвимостей;

д) уязвимостей кода.

Требования к уровню контроля ПО предъявляются в соответствии:

- с требованиями по безопасности информации, устанавливающими уровни доверия [14];

- с потенциалом нападения нарушителя, требуемого для эксплуатации уязвимостей и НДВ. Диапазоны конкретных значений потенциала нападений выделены в несколько уровней стойкости (безопасного ПО). Показано, как рассчитывается минимальный потенциал нападения (которому должно противостоять безопасное ПО). Потенциал рассчитывается при моделировании атак (включая тестирование на проникновение) для каждого сценария действий нарушителя при эксплуатации уязвимости. В методике показано, какой уровень (потенциал) нарушителя требуется для преодоления каждого уровня стойкости. Очевидно, что уровни возможностей нарушителей определены как Н1, Н2, Н3, Н4 в [20] в Приложении № 8.

Выявленные уязвимости и НДВ разработчик ПО должен устранить или принять меры для невозможности их эксплуатации. При этом меры по устранению уязвимостей и НДВ необходимо исследовать на:

- корректность внесенных в ПО исправлений (для устранения уязвимостей и НДВ);

- отсутствие новых уязвимостей, которые могли быть внесены в ПО в результате устранения уязвимостей.

Для каждой выявленной потенциально опасной функциональной возможности (опасного функционального объекта) должен быть проведен анализ й возможности возникновения угрозы безопасности информации и, если анализ подтвердит возможность возникновения угрозы (нарушения целостности, конфиденциальности или доступности ПО и обрабатываемых им данных), то потенциально опасная функциональная возможность (опасный функциональный объект) должна быть исключена разработчиком или он должен принять меры, обеспечивающие не возможность реализации выявленных угроз безопасности. информации.

Согласно Методике исследования по выявлению уязвимостей и НДВ ПО включают этапы:

1 этап: подготовка к проведению исследований по выявлению уязвимостей и НДВ ПО;

2 этап: проведение исследований по выявлению уязвимостей и НДВ ПО;

3 этап: оформление результатов исследований ПО на наличие уязвимостей и НДВ.

Рассмотрим работы на этих этапах.

1. Подготовка к проведению исследований ПО

1.1 Анализ документации

Должен быть проведен анализ программной и эксплуатационной документации, результатов оценки безопасности ПО от других организаций, базы данных угроз и уязвимостей.

По результатам анализа сведений о ПО могут быть выявлены потенциально опасные функциональные возможности ПО и ПО должно быть доработано.

1.2 Подготовка исследовательского стенда

Для обеспечения возможности проведения исследований ПО должен быть создан исследовательский стенд. В состав исследовательского стенда включается дистрибутив ПО и программно-аппаратное средство (в случае исследования встроенного ПО).

При подготовке стенда надо выполнить:

− настройку среды функционирования ПО, в которой оно должно (может) эксплуатироваться;

− установить и настроить ПО в соответствии с эксплуатационной документацией;

− разработать и применить меры безопасности для обеспечения целостности среды функционирования и возможности восстановления среды функционирования и самого ПО при проведении исследований.

− провести оценку (с помощью средств контроля защищенности информации от НСД) конфигурации и настроек ПО для выявления уязвимостей;

− провести проверку дистрибутива ПО и всех файлов установленного ПО на вирусы сертифицированными антивирусными средствами с актуальными на момент исследования базами данных.

2. Проведение исследований ПО

2.1 Экспертный анализ ПО

Экспертный анализ ПО включает:

− анализ документации на ПО и анализ всех доступных информационных источников со сведениями о ПО и его потенциальных уязвимостях, анализ результатов исследований ПО, полученных другими исследователями, с целью формирования перечня потенциальных уязвимостей, актуальных для ПО;

− анализ архитектуры ПО для выявления потенциально опасных функциональных возможностей ПО и уязвимостей в архитектуре (на уровне архитектуры) ПО;

− подтверждение потенциальных уязвимостей ПО путем сканирования уязвимостей и тестирования ПО на проникновение.

Экспертный анализ направлен не анализ того, как именно реализован и написан код того или иного блока, а направлен на выявление именно архитектурных уязвимостей, которые вызваны неправильным выбором структуры ПО или неправильным взаимодействием компонентов и блоков, из которых состоит ПО.

Основными методами, используемыми при проведении экспертного анализа, являются анализ документации, сканирование уязвимостей, моделирование атак, тестирование на проникновение.

В качестве источников информации для поиска информации об уязвимостях ПО можно использовать, например, банк данных угроз и уязвимостей ФСТЭК России (https://bdu.fstec.ru/).

Для выявления потенциальных уязвимостей ПО могут применяться программы-сканеры уязвимостей.

Для сканирования уязвимостей часто используются Nessus, Metasploit framework, OpenVAS, Сканер-ВС.

В ходе анализа архитектуры ПО, анализа доступных источников о ПО и сканирования уязвимостей должен быть сформирован перечень потенциальных уязвимостей (в том числе уязвимостей архитектуры), актуальных для ПО.

Для каждой выявленной уязвимости должны быть разработаны и проведены тесты на проникновение, в ходе которых подтверждается либо наличие либо отсутствие возможности эксплуатации этой уязвимости со стороны нарушителей.

2.2 Статический анализ ПО (SAST)

Статический анализ подразумевает анализ кода ПО без её непосредственного запуска на выполнение. Статический анализ проводится с помощью статических анализаторов кода.

Статический анализ ПО включает:

- построение модели ПО (синтаксического дерева, графа потоков управления, графа зависимостей по данным). Как правило, модель программы может иметь ряд упрощений и допущений, связанных с временными и ресурсными ограничениями. В связи с этим в результате статического анализа могут возникнуть как ложные обнаружения ошибок или программных конструкций, так и их пропуск в исследуемой программе;

− синтаксический анализ модели ПО путем сопоставления линейной последовательности лексем (в которую преобразуется исходный код в ходе лексического анализа) с заранее известным (формальным) набором последовательностей (сигнатур) лексем для ПО;

− семантический анализ исходного кода ПО - анализ путей выполнения кода ПО с учётом фактических параметров подпрограмм, значений переменных и т.п.;

− формальную верификацию исходного кода ПО, которая основана на строгом доказательстве отсутствия в ПО программных ошибок заданного класса.

Обычно при статическом анализе проводится синтаксический анализ, который может дополняться другими методами для:

- увеличения вероятности нахождения ошибок первого рода (межпроцедурный анализ)

- снижения вероятности ошибок второго рода (абстрактная интерпретация).

В качестве статических анализаторов обычно используются следующие:

− AppChecker для ПО на языках Java, C/C++/C#, PHP;

− Svace для ПО на языках C, C++, Java и C#;

− Coverity для ПО на языках Ruby on rails, Scala, PHP, Python, JavaScript, TypeScript, Java, Fortran, C, C++, C#, VB.NET;

− PVS-Studio для ПО на C, C++, C# и Java;

- Infowatch Appercut для ПО на 1C, C#, Java, JavaScript, SQL, PHP, Python, SAP Abap4 и LotusScript;

- Positive Technologies Application Inspector для ПО на PHP, Java, C#, JavaScript, Kotlin, VB.NET, Python, Objective-C, Swift, C/C++, Go, SQL (PL/SQL, T-SQL, MySQL);

- Solar inCode для ПО на Java, JavaScript, Scala, PHP, Python, Ruby, HTML5, PL/SQL, T/SQL, Java for Android, Swift, Objective C, C#, C/C++, VB 6.0, Delphi, ABAP, Solidity, Groovy, Kotlin.

Статические анализаторы могут автоматически находить следующие ошибки: а) переполнение буфера; б) целочисленное переполнение; в) неинициализированные («висящие») указатели; г) потерянные ссылки на выделенную область памяти; д) освобождение уже освобожденной области памяти; е) использование области памяти после ее освобождения; ж) разыменовывание неинициализированного указателя; з) неинициализированные переменные; и) недостижимый код; к) утечка информации (через сообщения об ошибках); л) использование операции присваивания вместо операции сравнения.

К недостаткам статического анализа обычно относят ошибки первого рода (пропуск ошибок в ПО) и ложные срабатывания (находятся ошибки, которые не являются ошибками).

Для ПО, используемого, для обработки документов, содержащих сведения, составляющие государственную тайну, используются формальные методы доказательства правильности функционирования ПО. Эти методы используют математические модели для описания спецификации программы и проверки ее правильности (соответствия спецификации).

Инструменты формальной верификации ПО можно разделить на три группы:

1. Инструменты проверки модели

Эти инструменты генерируют модель состояний ПО и проверяют, насколько набор состояний и их изменения соответствуют спецификациям.

2. SMT-решатели

Эти инструменты позволяют определить, выполнимы ли для анализируемого ПО полученные формулы (логика) первого порядка (исчисление предикатов).

3. Инструменты интерактивного доказательства теорем

Методы доказательства теорем делятся на две группы: 1) использующие аннотации, 2) использующие интерактивное доказательство.

Методы аннотирования используется для доказательства поведенческих свойств ПО. В ПО определяются предусловия и постусловия, по которым генерируются логические формулы для доказательств. Эти формулы подаются на вход SMT-решателю или доказываются в интерактивном режиме. В рамках интерактивного доказательства используется описание системы на одном из языков спецификаций, например, на Event-B.

2.3 Динамический анализ ПО (DAST)

Динамический анализ предусматривает выявление уязвимостей и НДВ в ходе анализа кода ПО в процессе непосредственного выполнения ПО. При проведении динамического анализа на вход программы подаются специально подобранные входные (тестовые) данные. При выполнении программы фиксируются результаты каждого теста (записываются входные и выходные данные теста, трассы выполнения программы и т.п.). Мерой оценки полноты проведения динамического анализа является степень покрытия кода набором тестов (выделяют покрытие операторов, покрытие ветвей, покрытие путей, покрытия всех точек входа и выхода функций и др.).

Для использования методов динамического анализа ПО требуется воссоздание его среды функционирования. В ходе создания среды функционирования ПО используются, как правило, штатные программно-аппаратные средства, используемые ПО. Если такие средства не могут быть получены, то возможно использование эмуляции аппаратного и программного обеспечения, необходимого для выполнения ПО в процессе исследований.

Динамический анализ ПО включает:

− фаззинг ПО для выявление уязвимостей на уровне архитектуры ПО и уязвимостей самого кода ПО; фаззинг основан на подаче на вход программы заведомо неправильных или случайных данных и фиксации нарушения штатного функционирования ПО в процессе обработки таких данных;

− фиксация и анализ соответствия трасс выполнения кода ПО тому порядку выполнения, который указан в программной документации. Результатом должно являться выявление опасных функциональных объектов в ПО и несоответствия связей между информационными и функциональными объектами ПО с теми связями, которые установлены в документации;

− анализ активности процессов взаимодействия программ, отслеживание помеченных данных (данных, зависящих от входа, т.е. чувствительных данных, таких как пароли, ключи шифрования и т.п.).

Основными методами динамического анализа являются фаззинг, мониторинг активности программ, отслеживание помеченных данных. Проведение динамического анализа осуществляется с использованием фаззера, монитора активности программ, отладчика, распаковщика, профилировщика.

Обычно фаззер находит такие ошибки как ошибки работы с памятью (в том числе переполнение буфера), утечка информации через неконтролируемую форматную строку, освобождение области памяти после ее освобождения, использование уже освобожденной области памяти и т.п.

Существует два типа фаззинга - мутационный и генерационный.

В рамках мутационного фаззинга на вход программы подаются случайные данные или неслучайные данные от запуска к запуску изменяют.

В рамках генерационного фаззинга входные данных не задаются явно. Вместо них используется описание структуры данных, по которому фаззер сам генерирует входные данные.

Большинство обнаруженных за последнее время ошибок в ПО, связанных с удаленным выполнением кода и повышением привилегий, найдены как раз при помощи фаззинга.

2.4 Ручной анализ ПО

Целью ручного анализа является поиск уязвимостей и НДВ в определенных фрагментах исходного кода:

- которые не были в достаточной мере охвачены статическим анализом, динамическим анализом (фаззингом, анализом трасс выполнения);
- которые написаны на языках, отличных от базового языка разработки ПО;
- которые являются самомодифицирующимся кодом.

Основным методом, используемым при проведении ручного анализа, является направленный ручной анализ участков кода.

Заканчивается ручной анализ кода подтверждением обычно потенциальных уязвимостей ПО.

3. Оформление результатов исследований ПО

По результатам исследований ПО на уязвимости и НДВ разрабатывается протокол проведения исследований. В протоколе должны быть отражены условия и результаты каждого исследования и результаты работы средств автоматизации исследования ПО, которые были использованы в процессах анализа ПО. Объем этих работ ранжирован по шести уровням контроля отсутствия НДВ.

Таким образом, **в данной работе показано, что есть два пути сделать ПО безопасным** - или использовать технологии безопасной разработки ПО, или искать ошибки и уязвимости ПО после завершения разработки ПО (с возможной последующей переделкой ПО).

Библиографический список

1. ГОСТ Р 56939-2016 «Защита информации. Разработка безопасного программного обеспечения. Общие требования». [Электронный ресурс]. - Режим доступа: https://protect.gost.ru/v.aspx?control=8&baseC=-1&page=0&month=-1&year=-1&search=&RegNum=1&DocOnPageCount=15&id=195653. - Дата доступа: 18.02.2023.

2. Приказ ФСТЭК России от 25 декабря 2017 г. № 239 «Об утверждении требований по обеспечению безопасности значимых объектов критической информационной инфраструктуры Российской Федерации». [Электронный ресурс]. - Режим доступа: https://fstec.ru/component/attachments/download/1880. - Дата доступа: 18.02.2023.

3. NIST SP 800-218 "Secure Software Development Framework (SSDF) Version 1.1: Recommendations for Mitigating the Risk of Software Vulnerabilities" («Фреймворк создания безопасного программного обеспечения: Рекомендации по снижению рисков программных уязвимостей»). [Электронный ресурс]. - Режим доступа: https://csrc.nist.gov/publications/detail/sp/800-218/final. - Дата доступа: 18.02.2023.

4. ГОСТ Р 15408-3-2013 «Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 3. Компоненты доверия к безопасности».

5. ГОСТ Р ИСО/МЭК 27034 «Информационная технология. Методы и средства обеспечения безопасности. Безопасность приложений».

6. ГОСТ Р ИСО-МЭК 18045—2013 «Информационная технология. Методы и средства обеспечения безопасности. Методология оценки безопасности информационных технологий».

7. ГОСТ Р 58142-2018 «Информационная технология. Методы и средства обеспечения безопасности. Детализация анализа уязвимостей ПО в соответствии с ГОСТ Р ИСО/МЭК 15408 и ГОСТ Р ИСО/МЭК 18045. часть 1 Использование доступных источников для идентификации потенциальных уязвимостей».

8. ГОСТ Р 58412-2019 «Разработка безопасного ПО. Угрозы безопасности информации при разработке ПО».

9. ГОСТ Р 58143-2018 «Информационная технология. Методы и средства обеспечения безопасности. Детализация анализа уязвимостей ПО в соответствии с ГОСТ Р ИСО/МЭК 15408 и ГОСТ Р ИСО/МЭК 18045. Часть 2 Тестирование проникновения.

10. ГОСТ Р 56545-2015 «Защита информации. Уязвимости информационных систем. Правила описания уязвимостей».

11. ГОСТ Р 56546—2015 «Защита информации. Уязвимости информационных систем. Классификация уязвимостей информационных систем».

12. ГОСТ Р 51188-98 «Защита информации. Испытания программных средств на наличие компьютерных вирусов».

13. Методика тестирования обновлений безопасности программных, программно-аппаратных средств.. Методический документ. Утвержден ФСТЭК России 28 октября 2022 г. [Электронный ресурс]. - Режим доступа: https://fstec.ru/component/attachments/download/3456. - Дата доступа: 18.02.2023.

14. Приказ ФСТЭК России от 02.06.2020 г. №76 Об утверждении требований по безопасности информации, устанавливающих уровни доверия к средствам ТЗИ и средствам обеспечения безопасности информационных технологий».

15. Информационное сообщение ФСТЭК России от 10 февраля 2021 г. № 240/24/647 «Об утверждении Методики выявления уязвимостей и недекларированных возможностей в программном обеспечении». [Электронный ресурс]. - Режим доступа: https://fstec.ru/component/attachments/download/2923. - Дата доступа: 18.02.2023.

16. Бегаев А.Н., Кашин С.В., Маркевич Н.А., Марченко А.А. Выявление уязвимостей и недекларированных возможностей в программном обеспечении – СПб: Университет ИТМО. - 2020. – 38 с. [Электронный ресурс]. - Режим доступа: http://books.ifmo.ru/file/pdf/2581.pdf. - Дата доступа: 18.02.2023.

17. Актуальные вопросы защиты информации //[Электронный ресурс]. - Режим доступа: https://lib.itsec.ru/articles2/focus/aktualnye-voprosy-zaschity-informatsii. - Дата доступа: 18.02.2023.

18. Сердечный, А.Л. Методика выявления уязвимостей и недекларированных возможностей в программном обеспечении. [Электронный ресурс]. - Режим доступа: https://ict.moscow/static/2-serdechnyj-a-l-4.pdf. - Дата доступа: 18.02.2023.

19. Падарян В.А. Разработка доверенного ПО. Вызовы и перспективы. [Электронный ресурс]. - Режим доступа: https://www.tbforum.ru/hubfs/TBF/Presentations2021_online/TBFOFFline/TBF 11-02-21 hall 3/ПАДАРЯН 1_Разработка доверенного ПО - Вызовы и перспективы.pdf. - Дата доступа: 18.02.2023.

20. Методика оценки угроз безопасности информации. Методический документ. Утвержден ФСТЭК России 5 февраля 2021 г. [Электронный ресурс]. - Режим доступа: https://fstec.ru/component/attachments/download/2919.. - Дата доступа: 18.02.2023.

УДК 004.65; ГРНТИ 20.23.17

СОВРЕМЕННОЕ ИСПОЛЬЗОВАНИЕ НЕРЕЛЯЦИОННЫХ БАЗ ДАННЫХ

А.Р. Силкина
Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, ari.silkina2015@yandex.ru

Аннотация. В работе рассматриваются нереляционные базы данных. Приводятся их классификация, достоинства и недостатки, сравнение их с реляционными базами данных, а также примеры их использования.
Ключевые слова: базы данных, системы управления базами данных(СУБД), реляционные базы данных, нереляционные базы данных (NoSQL), транзакция.

MODERN USAGE NON-RELATIONAL DATABASES

A.R. Silkina
Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, ari.silkina2015@yandex.ru

The summary. The paper considers relational databases. Their classification, advantages and disadvantages, their comparison with relational databases, as well as examples of their use are given.
Keywords: databases, database management systems, relational databases, relational databases (NoSQL), transaction.

Базы данных могут быть двух типов реляционные и нереляционные. Нереляционные базы данных иногда называются "NoSQL", что означает не SQL или не только SQL. Основное различие между ними заключается в том, как они хранят свою информацию.

Нереляционная база данных хранит данные не в форме таблиц и, как правило, более гибкая, чем реляционные базы данных.

Как работает нереляционная база данных?

Базы данных NoSQL используют различные модели данных для доступа к данным и управления ими. Они хорошо подходят для приложений, которые работают с большим объемом данных, нуждаются в высокой скорости реагирования и гибких моделях данных. Это получается реализовать за счет смягчения требований к данным.

Рассмотрим пример, в котором создается схема для базы данных книг.

В реляционной базе данных информация о книге разделяется на несколько составляющих и хранится в индивидуальных таблицах. Отношения между ними определяются ограничениями первичных и внешних ключей. В этом примере в таблице «Книги» имеются столбцы «Год издания», «Название книги» и «Номер издания», в таблице «Авторы» – столбцы «ID автора» и «Имя автора», а в таблице «Автор–Год издания» – столбцы «Автор» и «Год издания».

В нереляционных базах данных записи обычно хранятся в виде документов типа JSON. Для каждой книги значения «Год издания», «Название книги», «Номер издания», «Имя автора» и «ID автора» хранятся в одном документе. В такой модели данные оптимизированы для более легкой разработки и масштабируемости.

Типы нереляционных баз данных

БД, основанные на парах «ключ-значение». Такие базы данных поддерживают высокую разделимость и обеспечивают горизонтальное масштабирование, которое невозможно при использовании других типов баз данных. Примерами применения баз данных этого типа являются корзины интернет-магазинов.

Документоориентированные БД. В коде приложения данные, как правило, представлены как документы в формате JSON, из-за того, что это интуитивно понятно для разработчиков. Такие базы данных позволяют разработчикам сохранять и запрашивать данные с помощью тех же форм документов, которые они используют в коде приложения. Гибкий, слабоструктурированный, иерархический характер документов дает им возможность развиваться в соответствии с различными потребностями приложений. Документная модель хорошо работает в каталогах, профилях пользователей и системах управления контентом, в которых каждый документ неповторим и изменяется с течением времени.

Графовые БД. Такие базы данных делают проще разработку и использование приложений, работающих с наборами данных, обладающих сложными связями. Классическими примерами использования графовых баз данных являются социальные сети, биоинформатика и многие другие сферы, в которых необходимо коррелировать данные из различных источников.

БД в памяти. Часто в играх и рекламных приложениях мы можем наблюдать таблицы лидеров, функцию хранения сессий и аналитику в режиме реального времени. Для работы с такими данными скорость отклика должна быть в пределах нескольких милисекунд, при этом резкое возрастание трафика возможно в любой момент. Например, такие клиенты, как Tinder пользуются системами хранения данных в облачном хранилище памяти, а не на физическом диске.

Поисковые БД. Многие сервера формируют файлы журналов (логи), для того чтобы разработчикам было легче находить и ликвидировать неполадки. Существуют специализированные сервисы, которые используются для более понятного визуального представления и аналитики потоков данных, которые автоматически генерируются, практически в режиме реального времени, при помощи индексирования, агрегации полуструктурированных журналов и метрик, и поиска по ним.

Сравнение реляционных и нереляционных баз данных

В течение многих лет на первом месте в разработке приложений стояли реляционные базы данных, например, Oracle, SQL Server и PostgreSQL. Но ближе к концу нулевых годов стали заметно распространяться и другие модели данных. Они стали называться NoSQL, что можно перевести как не только SQL.

Существует множество типов нереляционных баз данных, в таблице 1 приведены основные отличия баз данных NoSQL от SQL.

Для чего можно использовать базы данных NoSQL?

Примером проекта, в котором используется нереляционная база данных является **проект rsreu-contests-system [3]**. Для решения была поставлена проблема очного проведения олимпиад по программированию, которое влечет за собой нагрузку на персонал организации, которая проводит олимпиаду – необходимо спланировать размещение конкурсантов по помещениям, обеспечить для них комфортные условия нахождения, подготовить бланки заданий, ответов, организовать проверку результатов.

Таблица 1 – Сравнение БД SQL и NoSQL

	Реляционные базы данных	Базы данных NoSQL
Подходящая нагрузка	Реляционные базы данных предназначены для использования в приложениях оперативной обработки транзакций (OLTP) и так же хорошо подходят для обработки запросов и отчетов (OLAP).	Нереляционные базы данных используются для работы с большим количеством вариантов доступа к данным, в том числе в приложениях с высокой скоростью реагирования. Поисковые базы данных предназначены для анализа полуструктурированных данных.
Модель данных	Требует жестко заданных таблиц, строк, столбцов, индексов, отношений между таблицами. Такая БД обеспечивает целостность данных в таблицах и приводит к уменьшению повторений в них.	Предоставляют различные модели данных, такие как документы, пары «ключ-значение» и графы, что помогает достичь высокой производительности и более легкой масштабируемости.
Требования ACID	Всегда выполняются требования ACID: атомарность, согласованность, изолированность, устойчивость. • Атомарность. Транзакция либо выполняется полностью, либо не выполняется вообще. • Согласованность. Как только транзакция завершается данные должны соответствовать четкой схеме базы данных. • Изолированность. Параллельные транзакции выполняются независимо друг от друга. • Устойчивость. Восстановление до последнего сохраненного состояния после различных сбоев в работе системы.	Базы данных NoSQL, как правило, смягчают жесткие требования свойств ACID для более гибкой модели данных, что позволяет использовать горизонтальное масштабирование.

Продолжение таблицы 1

	Реляционные базы данных	Базы данных NoSQL
Производительность	Производительность главным образом зависит от дисковой подсистемы. Для обеспечения максимальной производительности необходимо оптимизировать запросы, индексы и структуры таблицы.	Обычно зависит от возможностей аппаратного обеспечения, скорости реагирования сети и вызывающего приложения.
Масштабирование	Реляционные базы данных, как правило, масштабируются при помощи увеличения вычислительных возможностей аппаратного обеспечения.	Как правило поддерживают высокую разделимость с помощью специализированных шаблонов доступа с возможностью горизонтального масштабирования, в основе которой лежит распределенная архитектура. Это повышает пропускную способность и обеспечивает стабильную производительность практически без ограничений.
API	Запросы пишутся только на языке SQL.	Объектно-ориентированные API позволяют разработчикам приложений легко осуществлять запись и чтение структур данных. Приложения могут производить поиск по парам ключей и значений, наборам столбцов или полуструктурированным документам, содержащим атрибуты приложений.

Для автоматизации процесса проведения олимпиад по программированию было предложено разработать информационную систему (далее ИС) дистанционного проведения олимпиад по программированию с возможностью автоматической проверки. Подобные системы: Leetcode, Яндекс.Контест и др.

Одним из основных вопросов на начальной стадии разработки стал выбор СУБД для основного backend сервиса ИС. Для осуществления выбора были выделены следующие сущности предметной области: **пользователи** (участники олимпиад, организаторы), **организации**, на базе которых проводятся олимпиады (например, кафедра ВПМ РГРТУ), **заявки на создание организации в системе** (например, кафедра ЭВМ РГРТУ тоже захотела проводить олимпиады), **мероприятия** (непосредственно олимпиады, а также, подразумевалось, что преподаватели смогут проводить контроль усвоения материала на занятиях, связанных с программированием), **задачи** (олимпиадные), **информация об участниках мероприятия** (пользователи, прошедшие регистрацию на участие в конкретной олимпиаде), **решения задач** (решения олимпиадных задач, предоставленные участниками на конкретной олимпиаде), **информация о выполнении теста конкретной задачи** (решение каждой задачи тестируется

большим набором тестов, эта сущность должная хранить информацию о прохождении того или иного теста для конкретного решения конкретного участника), **апелляции** (попытка оспорить результаты автоматической проверки конкретного решения задачи).

После выделения сущностей было определено, что их можно вложить друг в друга, а также некоторые сущности ссылаются друг на друга (рис. 1).

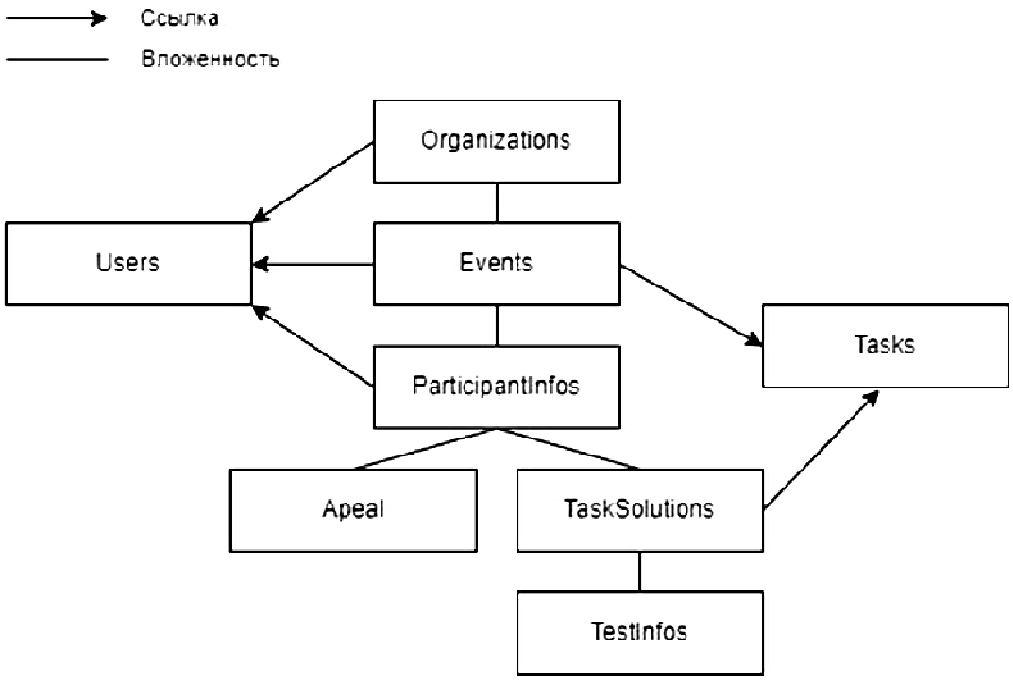


Рис. 1. Сущности предметной области

Для вложенных сущностей хорошо подходит NoSQL документоориентированная СУБД – Mongo Db. Для того, чтобы убедиться в этом построим ER диаграмму (рис. 2) для БД данной предметной области. Здесь мы имеем весьма сложную структуру, большое количество циклических связей между реляционными отношениями, вспомогательные таблицы для организации связи N:N.

В Mongo Db данные хранятся в виде документов без заранее заданной структуры (одно из отличий от реляционной модели). Каждый документ (элемент) может иметь собственную структуру. Структура документов в Mongo Db для предметной области отражена на рисунке 1. На этапе проектирования весьма удобно не задумываться о структуре документов, но в результате это может нарушать целостность данных. Однако, так как требования изначально были определены весьма размыто, отсутствие привязанности к структурированности данных имеет весомые преимущества.

При том, что о нормализации в Mongo Db речи идти не может, прикладные запросы к СУБД выполняются достаточно быстро; возможно построение индексов.

К «побочным эффектам» использования Mongo Db можно отнести необходимость осваивания администрирования, языка запросов, технологий на уровне прикладного кода.

По итогу после реализиции проекта стало понятно, что в данном примере использование нереляционных баз данных было не обоснованным и понесло большие затраты на обучение, которые того не стоили.

Однако, существуют и положительные примеры использования баз данных NoSQL, например, их использование существенно облегчает работу с логами из-за быстроты и динамических структур для хранения данных. Одним из примеров такой БД является Elasticsearch.

Заключение

Как правило, реляционные базы данных выбираются для решения повседневных задач, таких как учет кадров, материалов, книг и т.п., начисление заработной платы, в которых запросы сложны, а решения однообразны. Важным их отличием от нереляционных баз данных является то, что они масштабируются за счет увеличения нагрузки на конкретный сервер.

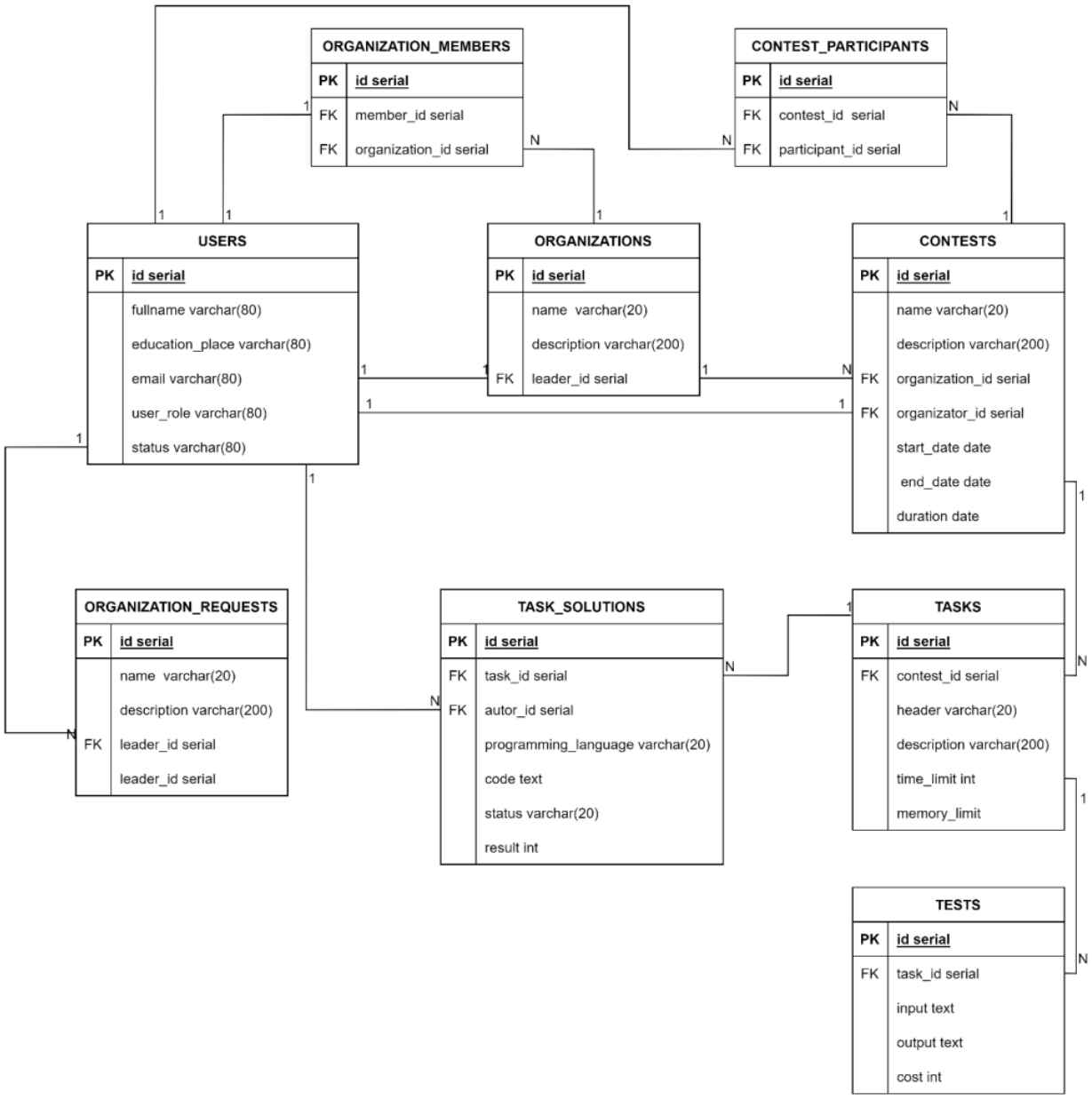


Рис. 2. ER-диаграмма

В то время как NoSQL масштабируются за счёт увеличения количества серверов, что особенно важно для работы с большими объемами данных.

В базах данных NoSQL хранится информация с различными атрибутами и полями, что позволяет решать задачи обработки недостаточно структурированных или абсолютно неструктурированных данных.

В связи со сказанным, нельзя точно определить какой из типов баз данных лучше. Нужно выбирать по необходимым для каждой конкретной задачи параметрам: способу масштабирования, типу данных, скорости доступа.

Библиографический список

1. Sanjay Sharma, Cassandra Design Patterns // Great Britain, Packt Publishing Ltd. -2014. – 88 с.
2. Садаладж, Фаулер, NoSQL. Методология разработки нереляционных баз данных// Киев, Диалектика. -2020. – 192 с.
3. Проект rsreu-contests-system [электронный ресурс] // URL https://github.com/maximastashkin/rsreu-contests-system

УДК 004.42; ГРНТИ 81.93.29

РАЗРАБОТКА КОМПЛЕКСНОГО ПРОГРАММНОГО СРЕДСТВА УПРАВЛЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТЬЮ НА ПРЕДПРИЯТИИ

Д.Р. Бабаев*, С.А. Ларинский**
Рязанский государственный радиотехнический университет имени В.Ф. Уткина,
Российская Федерация, Рязань
**babaev.daniil.r@gmail.com*
***s.larinskiy@yandex.ru*

Аннотация. В работе рассматриваются проблемы обеспечения информационной безопасности на предприятиях, а также программное решение, позволяющее автоматизировать процессы электронного документооборота, касающегося вопросов информационной безопасности.
Ключевые слова: информационная безопасность, web-разработка, документооборот, Python.

DEVELOPMENT OF A COMPREHENSIVE SOFTWARE TOOL INFORMATION SECURITY MANAGEMENT AT THE ENTERPRISE

D.R. Babaev*, S.A. Larinskiy**
Ryazan State Radio Engineering University named after V.F. Utkin,
Russia, Ryazan
**babaev.daniil.r@gmail.com*
***s.larinskiy@yandex.ru*

The summary. The paper discusses the problems of ensuring information security on enterprises, as well as a software solution that allow automating the processes of electronic document management related to information security issues.
Keywords: information security, web-development, document management, Python.

Информационная безопасность на любом предприятии обеспечивается не только техническими, но и организационными мерами, включающими в себя создание ряда документов. В тех случаях, когда в организации происходит активное развертывание новых автоматизированных информационных систем, возникает необходимость их документационного сопровождения, в том числе касающегося информационной безопасности, в результате чего процессы документооборота начинают требовать больших временных ресурсов. Данная ситуация может привести подразделения информационной безопасности и организацию в целом к состоянию, когда безопасность предприятия обеспечивается номинально: составляются необходимые документы, но средства защиты информации настраиваются неверно, мероприятия по оценке эффективности не проводятся

либо проводятся условно.

На практике возникает необходимость создания средства автоматизации процессов документооборота для повышения качества работы службы информационной безопасности организации. Разрабатываемое средство должно осуществлять свои функции с соблюдением требований безопасной разработки [1], так как обрабатываемые им документы относятся к чувствительным, позволяющим злоумышленнику в случае их утечки выстроить вектор атаки на организацию. Архитектура средства автоматизации должна обеспечивать одновременную работу нескольких пользователей, обработку, хранение и группировку документов, сведений о сотрудниках, информационных системах, к которым подключается организация, собственных информационных системах, а также предусматривать подсистему управления задачами сотрудников. Помимо этого, программное средство должно быть совместимо с отечественными операционными системами [2].

Выбор архитектуры и инструментов

Для организации взаимодействия комплекса и пользователя была выбрана архитектура «клиент-сервер» с использованием веб-приложения. Она обеспечивает кроссплатформенность и низкие системные требования для машин пользователей, а также возможность одновременного подключения и работу с системой большого количества пользователей.

В качестве основного языка программирования был выбран язык программирования Python. Python обладает большим количеством достоинств, к которым относится лаконичность языка, а также большое количество библиотек и фреймворков для разнообразных целей. На базе языка программирования Python организацией «Django Software Foundation» был создан свободно распространяемый фреймворк для веб-приложений Django, который был использован для разработки данного комплекса. Python является интерпретируемым языком, что означает, что программы выполняются дольше, по сравнению с компилируемыми языками, однако современные интерпретаторы Python выполняют команды довольно быстро, приближаясь по быстродействию к компилируемым. К тому же использование фреймворка Django с помощью встроенных функций позволяет защититься от большого количества атак: защита от межсайтовых скриптов (XSS), защита от подделки межсайтовых запросов (CSRF), защита от SQL-инъекций, защита от Clickjacking и другие [3]. К тому же использование Django позволяет развернуть HTTPS на своем сайте.

Функциональный модуль генерации документов

Одной из основных задач комплекса является автоматизация процессов создания документов. Прикладной пакет docxtpl для языка программирования Python позволяет создавать шаблоны файлов формата «docx» и заполнять их с помощью разметки полей. На вход функции создания пакета поступает пути к входному шаблону и конечному документу, а также список размеченных полей и значений для их заполнений.

Одной из проблем на стадии разработки стала избыточность входных данных для работы генератора. Для создания одностраничного документа программе на вход приходилось передавать около 80 переменных, содержащих словосочетания для заполнений шаблона. Причем каждые 5-6 словосочетаний отличались лишь склонениями главного слова. Данная ситуация привела к увеличению сложностей с обслуживанием комплекса. Например, чтобы добавить в шаблоны поле, которое заполнялось бы значением «информационная система персональных данных», требовалось бы создать 12 записей: 6 склонений с заглавной буквы и 6 склонений с маленькой буквы.

Решением данной проблемы стало использование морфологического анализатора pymophy. Данный анализатор использует базу морфем, позволяющую определить наиболее вероятную форму слова и произвести его склонение. Однако у данного решения есть свои недостатки.

Во-первых, pymorhy может анализировать и склонять лишь одно слово, причем после склонения заглавная буква утрачивалась. На вход модулю можно подавать массив слов, но это приведет к искажению словосочетаний. Так, например, словосочетание «начальник отдела» при склонении в родительный падеж будет заменено на «начальником отделом».

Во-вторых, анализатор может с определенной долей вероятности неверно определять формы некоторых слов. Например, слово «техник» анализатор наиболее вероятно определяет как родительный падеж множественного числа от слова «техника» и наименее вероятно, как именительный падеж единственного числа слова «техник». В случае необходимости склонения словосочетания «техник по защите информации» в творительный падеж будет выдано «техниками по защите информации» вместо «техником по защите информации».

Для использования анализатора pymorhy была произведена разработка промежуточной функции. Во всех словосочетаниях, используемых в научном стиле речи с главным существительным словом, имена прилагательные идут до главного слова, а подчиненные существительные после главного [4], причем при склонении главного слова форма меняется только у зависимых прилагательных. Именно поэтому разработанная функция определяет положение первого имени существительного в словосочетании и производит склонение всех слов до него включительно. Помимо этого, перед склонением имени существительного функция проверяет его соответствие нормальной форме предполагаемого слова. Так как в базе данных комплекса все словосочетания указываются в именительном падеже, естественно утверждать, что существительное будет в нормальной форме.

Связка модуля docxtpl с модулем pymorhy посредством разработанной функции позволила снизить количество входных переменных в 5 раз и избавиться от необходимости внесения и хранения в базе данных разных склонений одной записи.

Функциональный модуль управления задачами и уведомлениями

Разработанный комплекс обеспечивает управление задачами. Пользователь системы, обладающий необходимыми правами начальника подразделения, может создавать новые задачи своим подчиненным, устанавливать сроки их выполнения, необходимые отчетные документы и контролировать процесс их выполнения. Сотрудник подразделения при назначении новой задачи получает уведомление об этой задаче на странице комплекса, а также получает возможность прикреплять файлы к задаче и менять ее статус на статусы «В работе», «Ожидание ответа», «Отложено» или «Выполнено». Когда задаче устанавливается статус «Выполнено», она отправляется на согласование постановщику задачи, который может либо завершить ее, либо вернуть на доработку.

Модуль управления уведомлениями обеспечивает оповещение пользователя о важных событиях. Все предстоящие уведомления, связанные с задачами и сроками документов, хранятся в единой таблице. Системный таймер каждую минуту инициирует событие, по которому обработчик производит проверку данной таблицы. Если текущее время больше или равно времени предстоящего уведомления, то запись из общей таблицы перемещается в таблицы прикрепленных пользователей, после чего происходит отображение нового уведомления на странице. В случаях, когда уведомление необходимо доставить пользователю мгновенно, запись помещается в таблицу пользователя минуя общую таблицу предстоящих уведомлений.

Модуль «Классификатор»

На практике часто возникает необходимость определения уровня защищенности персональных данных информационной системы персональных данных и (или) класса защищенности информационной системы. Для удобного и быстрого решения такой задачи в комплексе предусмотрен специальный модуль «Классификатор». Он представляет из себя набор переключателей для определения уровня или класса защищенности. Первым пунктом пользователю предлагается выбрать, что он будет классифицировать (по умолчанию выбран «ГИС», т.е. класс защищенности информационной системы). На основании этого выбора на странице будут отображаться переключатели для соответствующей классификации.

В соответствии с действующим законодательством [5, 6] должны быть предусмотрены следующие переключатели:

1) определение уровня защищенности персональных данных:
   а) категория персональных данных;
   б) категория субъектов;
   в) количество субъектов;
   г) актуальный тип угроз;
2) определение класса защищенности:
   а) уровень значимости информации (конфиденциальность, целостность, доступность);
   б) масштаб информационной системы.

Для более рационального использования серверных ресурсов введенные на страницу данные не отправляются на сервер с помощью HTTP-метода POST, а обрабатываются с помощью скрипта страницы, написанного на языке программирования JavaScript.

На рисунке 1 представлен внешний вид модуля «Классификатор».

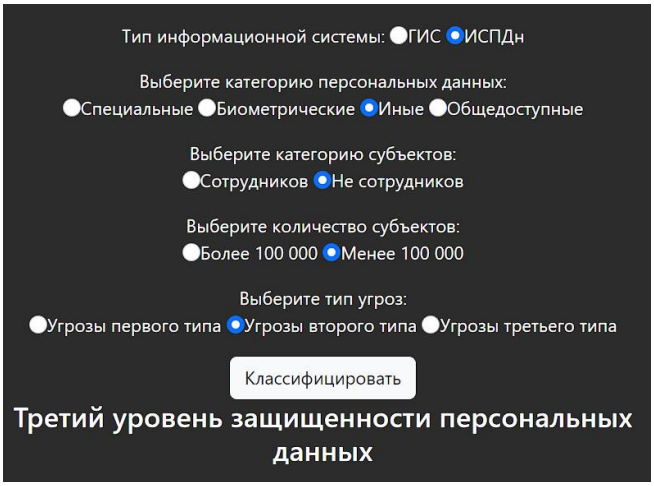


Рис. 1. Модуль «Классификатор»

Операции над данными

В задачи комплекса входит обработка большого количества видов данных. Данные необходимо вводить, сохранять, изменять, выводить по запросу пользователя, а также отправлять для обработки в другие модули. Для осуществления всех этих действий у комплекса должна быть база данных, в которой будут храниться все данные для обеспечения автоматизированного обеспечения информационной безопасности и ее управления.

Фреймворк Django при создании проекта автоматически создает базу данных, прописывает метаданные для взаимодействия с ней и производит ее первоначальную инициализацию системными данными. По умолчанию в Django используется свободно

распространяемая компактная встраиваемая система управления базами данных SQLite3, но благодаря своей удобности и хорошей совмести с фреймворком разработчиками было принято решение оставить ее как основную систему управления базами данных комплекса.

В комплексе обрабатываются следующие виды данных:

1) сведения об автоматизированных рабочих местах организации или информационных системах – представляют из себя название внутренней информационной системы, уровень (класс) защищенности информационной системы, номер помещения, в котором она располагается, состав, связанные файлы, которые необходимо загружать, список ответственных лиц, список информационных систем, к которым осуществляется подключение, если оно осуществляется, список установленного программного обеспечения и список вспомогательных технических средств и систем;

2) сведения об ответственных лицах – представляют из себя ФИО сотрудника, его должность с указанием отдела и электронную почту сотрудника, а также идентификаторы и хеш-суммы паролей;

3) сведения об информационных системах – представляют из себя название информационной системы, уровень (класс) защищенности информационной системы, категории обрабатываемых данных;

4) сведения о программном обеспечении – представляют из себя наименование программного обеспечения, производитель и сведения о сертификате;

5) сведения о сертификатах на программное обеспечение – представляют из себя номер сертификата, дата внесения в реестр, срок действия сертификата.

сведения о поставленных задачах – представляют из себя наименование задачи, сотрудника, сформировавшего задачу, сотрудника (сотрудников), для которого (которых) была поставлена задача, время, до которого требуется выполнить задачу.

По итогу используются шесть таблиц, пять из которых связаны отношениями. Таблица с собственными информационными системами связана с таблицами ответственных лиц, программного обеспечения и внешних информационных систем связана отношениями ManyToMany, то есть «многие ко многим», что не удовлетворяет даже первой нормальной форме базы данных. Однако данная возможность реализуется с помощью встроенных функций фреймворка Django, который автоматически создает в базе данных дополнительные таблицы для безопасной реализации отношений ManyToMany между таблицами, что приводит базу данных к первой нормальной форме, а приведение базы данных к последующим нормальным формам возлагается на архитектора базы данных. В создаваемых таблицах находятся соответствия между первичным ключом первой таблицы и первичным ключом второй таблицы для данного поля ManyToMany. Фреймворк также автоматически прописывает системные функции для взаимодействия со всеми таблицами.

В результате разработчик создает модель базы данных, представленной на рисунке 2, а фреймворк Django автоматически приводит ее к виду, представленному на рисунке 3.

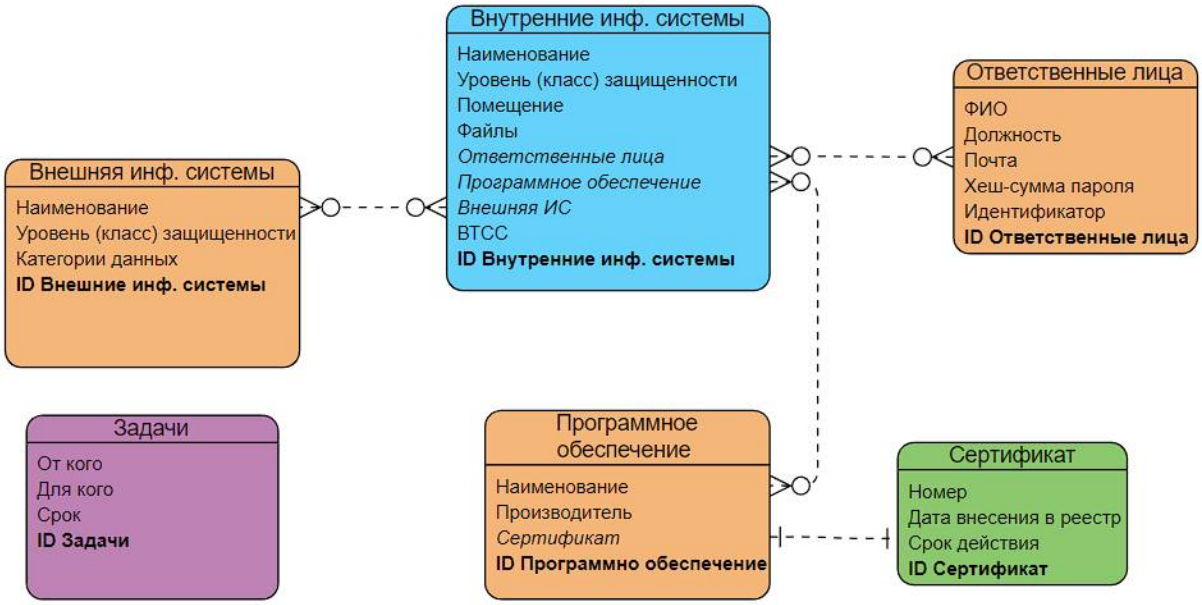


Рис. 2. Модель базы данных, создаваемых пользователем

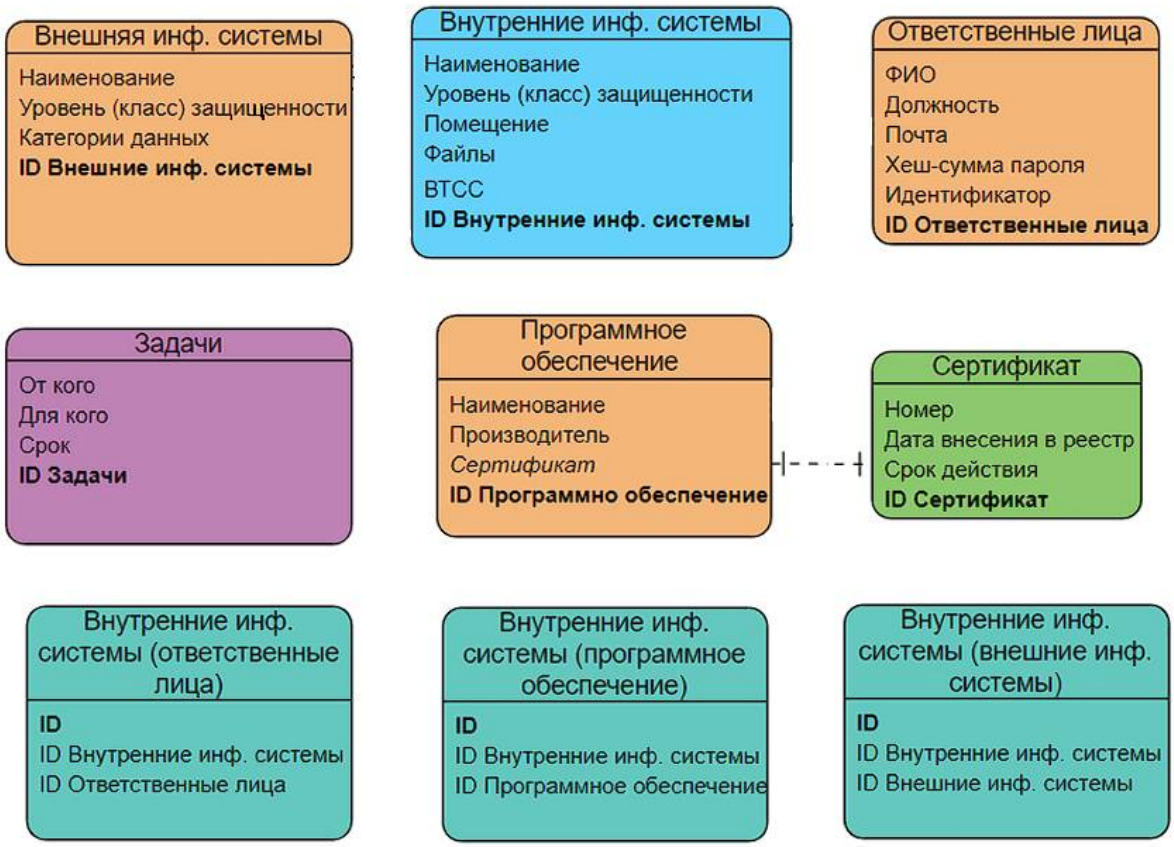


Рис. 3. Реально создаваемая модель базы данных

Таблица со сведениями о сертификатах заполняется автоматически при нажатии

кнопки «Обновить базу данных сертификатов». После нажатия на нее с официального сайта ФСТЭК России происходит скачивание файла формата *.csv с необходимыми сведениями о сертификатах. Далее таблица с сертификатами очищается и вновь заполняется с помощью данных из csv-файла. При необходимости есть возможность ввода сертификатов вручную.

Защита компонентов комплекса от несанкционированного доступа

Так как комплекс предназначен для служебного пользования, возникает необходимость в защите информации, обрабатываемой в нем. Так, для защиты трафика комплекса и пользователя используется протокол защиты транспортного уровня TLS [7]. Доступ к страницам сайта разрешается только после авторизации пользователя, который создается в системе администратором программного комплекса.

Выводы

Разработанный программный комплекс позволит руководителям организации и ответственным лицам оперативно контролировать информационную безопасность на своем предприятии, автоматизировать создание документов, оптимизировать документооборот в целом, контролировать работу подчиненных, а также обеспечивать взаимодействие между сотрудниками из разных отделов, увеличивать производительность их совместной работы благодаря оперативному доступу к документам и необходимым для работы сведениям.

Библиографический список

1. ГОСТ Р 56939-2016 Защита информации. Разработка безопасного программного обеспечения. Общие требования. – Москва: Стандартинформ, 2018.

2. Указ Президента РФ от 30 марта 2022 г. № 166 «О мерах по обеспечению технологической независимости и безопасности критической информационной инфраструктуры Российской Федерации».

3. Django Software Foundation. Django Documentation Release 4.1.8.dev20230215082617. – 2023.

4. Бабайцева В. В., Максимов Л. Ю. Современный русский язык. Часть 3. Синтаксис. Пунктуация. – Москва: Издательство «Просвещение», 1987.

5. Федеральная служба по техническому и экспортному контролю. Приказ от 11.02.2013 № 17 «Об утверждении требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах».

6. Постановление Правительства Российской Федерации от 01.11.2012 г. № 1119 «Об утверждении требований к защите персональных данных при их обработке в информационных системах персональных данных».

7. Мартыненков И. В. Прикладная дискретная математика. Основные этапы развития криптографических протоколов SSL/TLS и IPsec. – Москва: 2021.

УДК 004.056.55; ГРНТИ 81.93.29

ПРОТОКОЛ IPLIR И ЕГО ПРИМЕНЕНИЕ В СЕТЯХ КОМПЛЕКСА VIPNET

Д.Р. Бабаев*, С.А. Ларинский**
Рязанский государственный радиотехнический университет имени В.Ф. Уткина,
Российская Федерация, Рязань
**babaev.daniil.r@gmail.com*
***s.larinskiy@yandex.ru*

Аннотация. В данной статье описывается протокол IPlir, проводятся оценки его эффективности и демонстрируется его перехват и анализ трафика атакующим.
Ключевые слова: перехват трафика, сетевые технологии, виртуальные частные сети, iplir, ViPNet.

THE IPLIR PROTOCOL AND ITS APPLICATION IN THE NETWORK OF THE VIPNET COMPLEX

D.R. Babaev*, S.A. Larinskiy**
Ryazan State Radio Engineering University named after V.F. Utkin,
Russia, Ryazan
**babaev.daniil.r@gmail.com*
***s.larinskiy@yandex.ru*

Abstract. This article describes the IPlir protocol, evaluates its effec-tiveness, and demonstrates its interception and analysis of traffic by attacker.
Keywords: traffic interception, network technologies, virtual private networks, iplir, ViPNet.

ViPNet – это флагманская технология, выпущенная на рынок компанией InfoTecs в 1992 году, позволяющая организовывать виртуальные частные сети для защищённой передачи данных. С момента выхода прототипа продукта прошло несколько десятков лет, в течение которых технология модифицировалась и расширялась. В настоящее время данным решением пользуются многие государственные и коммерческие учреждения для создания защищённых сетей передачи данных (ЗСПД). Одним из примеров таких сетей является ЗСПД №3608 Федерального реестра документов об образовании для обмена информацией о выданных документах.

Теоретические сведения о протоколе IPlir

Фундаментом технологии ViPNet является стандартизированный протокол безопасности сетевого уровня IPlir (IP layer interconnection robust), обеспечивающий конфиденциальность и имитостойкость данных циркулируемых в сетях связи с использованием протокола TCP/IP. Данный протокол также утвержден в качестве рекомендации по стандартизации приказом Росстандарта [1].

Пакет IPlir [2] состоит из заголовка IP, заголовка UDP (используется при дополнительной инкапсуляции сообщения IPlir в сообщении UDP) и самого сообщения IPlir, структура которого изображена на рисунке 1.

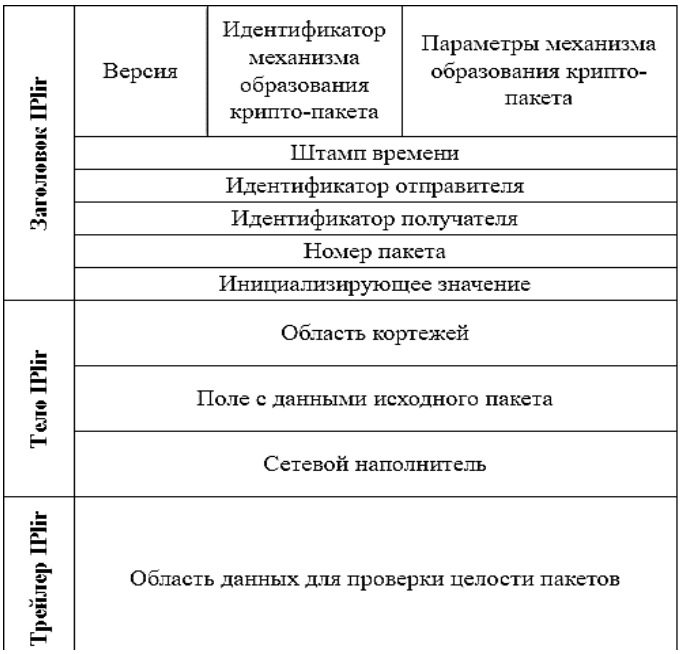


Рис. 1. Структура пакета iplir

IPlir сообщение включает в себя заголовок IPlir, содержащий служебную открытую информацию о параметрах защиты исходного пакета, тело IPlir с данными, трейлер IPlir, включающий в себя имитовставки, информацию для проверки целостности.

Отличительной особенностью протокола IPlir является выработка уникальных ключей шифрования, сквозной и транзитной имитозащиты для каждого пакета. В ходе криптографической защиты передаваемой информации применяются блочные шифры, основанные на государственных стандартах «Магма» или «Кузнечик» [3].

Подготовка к оценке эффективности протокола IPlir

Для проверки криптографической стойкости пакетов протокола IPlir была создана и настроена локальная сеть [5] из виртуальных машин на базе ПО «VMware». На данной сети развернут комплекс сети ViPNet структуры «Клиент-Координатор-Администратор», изображенной на рисунке 2.

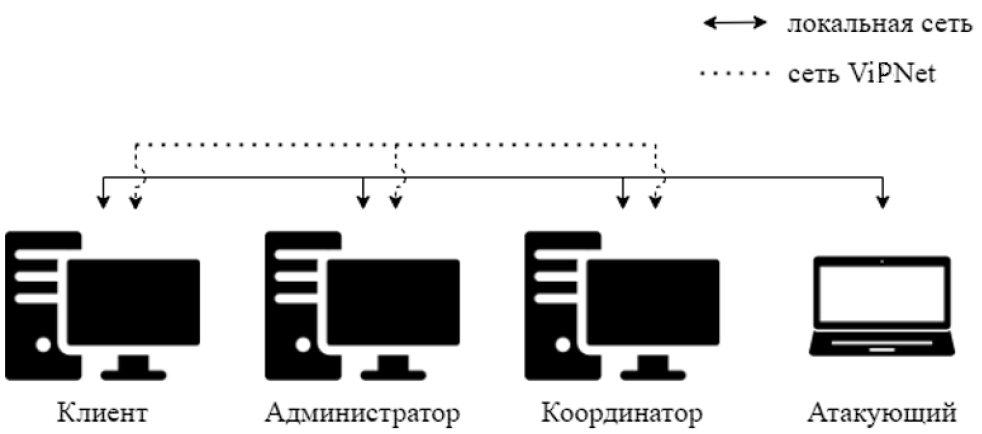


Рис. 2. Структура виртуальной сети

Администратор организует логическую инфраструктуру сети ViPNet, осуществляет мониторинг и выдачу ключевой информации. ЭВМ администратора содержит 2 программных модуля: удостоверяющий ключевой центр и центр управления сетью. Также на нём имеется ПО «ViPNet Client» для ведения защищённого обмена файлами на платформе ОС «Windows 7».

Координатор осуществляет регистрацию передаваемых пакетов, а также выполняет функции сетевого шлюза локальной сети. ЭВМ координатора включает в себя ПО «ViPNet Coordinator 4» на базе ОС «Astra Linux».

Клиент принимает участие в информационном обмене, шифрует и расшифровывает передаваемые пакеты. В состав ПО ЭВМ Клиента входит «ViPNet Client» для платформы ОС «Windows 7».

Атакующий имеет доступ в локальную сеть функционирования комплекса ViPNet, однако ключевая информация, как и доступ к самой частной виртуальной сети, у него отсутствует.

Так как атакующий имеет доступ к сети, у него имеется возможность перехватывать пакеты, их модифицировать и перенаправлять. При оценке параметров протокола IPlir будет использоваться программа анализатор трафика «Wireshark 3.6.5», позволяющая осуществлять мониторинг передаваемых в сети пакетов, на платформе свободной ОС «Kali Linux 2022.1». Стоит отметить, что в ПО ViPNet предусмотрена утилита «iplir view» для мониторинга трафика в сети, однако с позиции атакующей стороны она не является доступной, так как доступ к ней ограничен и её функционал не позволяет просматривать содержимое пакетов.

Атакующий для получения требуемого объёма данных для анализа ведёт мониторинг обмена пакетами в сети с произвольного момента времени её функционирования и на протяжении неограниченного периода.

Анализ пакетов протокола IPlir атакующим

Главной задачей атакующего является перехват трафика, циркулируемого в сети, и последующее выделение из него полезной информации.

Для этого в ПО «Wireshark» в качестве прослушиваемого интерфейса указывается интерфейс локальной сети «eth0». Для того, чтобы отфильтровать пакеты и исключить ненужные, применяется фильтр «(ip.src == 192.168.240.25 || ip.src == 192.168.240.23)&&!udp». В секции «ip.src» указаны IP-адреса, между которыми циркулирует шифрованный трафик (нет стандартной информации от ПО «Wireshark» о протоколе пакета). Секция «!udp» исключает из отображения UDP пакеты, распознанные анализатором. В итоге на вывод поступает список пакетов, фрагмент которого указан на рисунке 3.

No.	Time	Source	Destination	Protocol	Length	Info
76	36.297020225	192.168.24…	192.168.24…	IPv4	114	Unknown (241)
80	39.841573979	192.168.24…	192.168.24…	IPv4	308	Unknown (241)
81	39.842293682	192.168.24…	192.168.24…	IPv4	310	Unknown (241)
127	60.354761452	192.168.24…	192.168.24…	IPv4	114	Unknown (241)
130	60.886728343	192.168.24…	192.168.24…	IPv4	123	Unknown (241)
133	60.889098591	192.168.24…	192.168.24…	IPv4	111	Unknown (241)
134	60.891244133	192.168.24…	192.168.24…	IPv4	190	Unknown (241)
136	60.902079342	192.168.24…	192.168.24…	IPv4	116	Unknown (241)
138	60.915278944	192.168.24…	192.168.24…	IPv4	114	Unknown (241)
154	68.343261253	192.168.24…	192.168.24…	IPv4	308	Unknown (241)
155	68.344063616	192.168.24…	192.168.24…	IPv4	310	Unknown (241)

Рис. 3. Окно вывода программы «Wireshark» с перехваченными пакетами

После перехвата трафика атакующему требуется более детально ознакомиться с содержимым отдельных пакетов. В дальнейшем будет проведён детальный анализ пакета №130 (рис. 4).

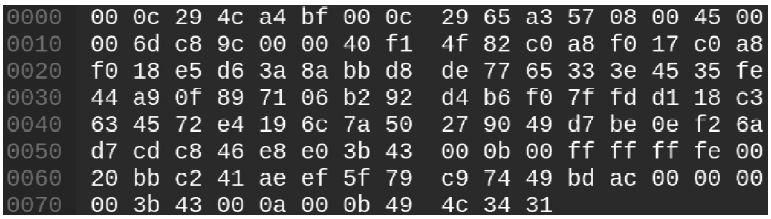


Рис. 4. Окно вывода программы «Wireshark» с содержимым перехваченного пакета №130

В качестве примера в таблице 1 представлены данные, полученные в ходе перехвата содержания пакета №130 TCMP Echo-Request («пинг»-запрос с ЭВМ Администратора на ЭВМ Клиента). Стоит отметить, что при изменении характера передаваемой информации, маршрутов передачи и других параметров, влияющих на изначальный нешифрованный пакет, общая структура пакета IPlir не изменяется.

Таблица 1 – Расшифровка содержимого пакета IPlir

Содержимое пакета	Назначение	Расшифровка
1	2	3
0x 00 0c 29 4c a4 bf	Адрес назначения	00:0c:29:4c:a4:bf
0x 00 0c 29 65 a3 57	Адрес источника	00:0c:29:65:a3:57
0x 08 00	Версия протокола IP	IPv4
0b 01 00 01 01	Версия и длина заголовка	Версия – 4, длина 20 байт
0b 00 00 00 00	Поля DSCP и ECN*	Без приоритета, без ECN
0x 00 6d	Общая длина	109 байт
0x c8 9c	Идентификатор пакета	51356
0x 00 00	Флаги и смещение сегмента	Без флагов, без смещения
0x 40	Время жизни пакета	64 перехода
Содержимое пакета	Назначение	Расшифровка
0x f1	Протокол	IP/241 (IPlir)
0x 4f 82	Контрольная сумма заголовка	0x 4f 82
0x c0 a8 f0 17	IP-адрес отправителя	192.168.240.23
0x c0 a8 f0 18	IP-адрес получателя	192.168.240.24
0x …	Поле данных	Содержимое пакета

В головную часть пакета помещена информация, необходимая для его передачи в сети [4]. Эти данные не шифруются для сохранения возможности передачи пакетов между сетями. С помощью головной части атакующий может узнать лишь общие сведения о передаваемой информации: её отправителя, получателя, обобщённую структуру сообщения и инкапсулированный в пакете протокол. Как будет разъяснено ниже, раскрытие используемого инкапсулированного протокола не даст атакующему дополнительных возможностей для продвижения анализа.

Наибольший интерес для злоумышленника представляет поле данных, в котором находятся тело IPlir, так как именно в нём находится полезная нагрузка, требуемая атакующему. В результате анализа пакетов с различным целевым назначением, были выявлены характерные структурные части, которые сохраняют свой порядок и чьи значения предсказуемы, независимо от времени передачи, содержания, отправителя и получателя пакета. Эти данные представлены в таблице 2.

Таблица 2 – Содержимое поля данных IPlir

Содержимое блока	Предполагаемое назначение	Расшифровка
1	2	3
0x …	Блок с данными переменной длины	Полезная нагрузка
0x 3b 43 00 0b 00	Информация об отправителе	Изменяет значение 4-ого байта при изменении отправителя
0x ff ff ff fe 00 20	Служебная информация	Неизменная часть
0x …	Блок с данными переменной длины	Полезная нагрузка
0x 00 00 00 00	Наполнитель	Неизменная часть, не несёт информации
0x 3b 43 00 0b 00	Информация об отправителе	Изменяет значение 4-ого байта при изменении отправителя
0x 0b 49 4c 34 31	Служебная информация, признак конца пакета	Неизменная часть

Атакующий с помощью этих данных не может получить новых полезных сведений. Искомая нагрузка зашифрована симметричным ключом шифрования с применением алгоритмов блочного шифрования и требует наличия ключевой информации сети ViPNet, а также входных параметров шифрования. Грубое же дешифрование перебором требует больших вычислительных мощностей, которыми злоумышленник не обладает.

Также атакующий не может провести модификацию или подмену передаваемых пакетов. Их достоверность обеспечивается использованием добавляемых к сообщениям имитовставок, получаемых применением симметричного криптографического метода. В результате модификации сообщения значения имитовставок становятся ложными, что свидетельствует о недостоверности сообщения. Также в комплексе ViPNet вместе с ключевой предусмотрена и справочная информация, содержащая данные о существующих связях между элементами сети.

Выходные данные, получаемые атакующим

В результате анализа циркулируемых в сети ViPNet пакетов протокола IPlir атакующий по перехваченным данным может [7] раскрыть состав анализируемой сети, исходя из содержимого пакетов и интенсивности их поступления, типы сообщений по их характерным чертам обмена, например, порядком обмена между ЭВМ или продолжительностью обмена.

Данная информация не является критической для безопасности сети, так как критерии информационной безопасности соблюдаются, пакеты, циркулируемые в сети, остаются достоверными и конфиденциальными.

Вывод

На сегодняшний день развитие информационного сообщества происходит с колоссальными скоростями. Вместе с этим растёт и количество организаций, занимающихся обработкой персональных данных и конфиденциальных документов. Обеспечение защищённой работы с этой информацией – ключевая задача для всех представителей не только сферы информационных технологий, но и иных сфер, занимающихся обработкой данных конфиденциального содержания. Это всё невозможно без протоколов шифрования, одним из которых и является IPlir. Сеть ViPNet продемонстрировала свою устойчивость к атакам внешних нарушителей и смогла обеспечить достоверность и конфиденциальность передаваемой информации.

Библиографический список

1. Об утверждении рекомендаций по стандартизации Российской Федерации: приказ Министерства промышленности и торговли Российской Федерации от 16 декабря 2020 г. №1325-ст // URL: https://tc26.ru/upload/medialibrary/2db/Приказ%20IPlir.pdf.

2. Р 1323565.1.034-2020. Информационная технология. Криптографическая защита информации. Протокол безопасности сетевого уровня: введ. 2021-04-01. – М. // URL: https://docs.cntd.ru/document/603366553.

3. ГОСТ 34.12-2018. Информационная технология. Криптографическая защита информации. Блочные шифры: введ. 2019-06-01. – М. // URL: https://docs.cntd.ru/document/1200161708#7D20K3.

4. RFC 791. Internet Protocol. Darpa internet program. Protocol specification: сентябрь 1981 г. // URL: https://datatracker.ietf.org/doc/html/rfc791 (дата обращения 06.05.2022) – P. 11.

5. Акинина Л.Н., Попов В.Б., Перехрест Р.Д. Программно-аппаратные комплексы ViPNet и их использование в корпоративных сетях // Научный вестник Крыма. 2016. №3 (3). URL: https://cyberleninka.ru/article/n/programmno-apparatnye-kompleksy-ViPNet-i-ih-ispolzovanie-v-korporativnyh-setyah – С. 9-10.

6. Прудников А.И., Шахов В.Г. Особенности использования технологии ViPNet для защиты информации в корпоративных сетях // ОНВ. 2012. №2 (110). URL: https://cyberleninka.ru/article/n/osobennosti-ispolzovaniya-tehnologii-ViPNet-dlya-zaschity-informatsii-v-korporativnyh-setyah – С. 272.

7. Федорченко Е.В., Новикова Е.С., Гайфулина Д.А., Котенко И.В. Построение профиля атакующего на основе анализа сетевого трафика в критических инфраструктурах // Системы управления, связи и безопасности. 2021. №6. URL: https://cyberleninka.ru/article/n/postroenie-profilya-atakuyuschego-na-osnove-analiza-setevogo-trafika-v-kriticheskih-infrastrukturah – С. 79-83.

УДК 004.91; ГРНТИ 81.93.29

ПОДТВЕРЖДЕНИЕ ПОДЛИННОСТИ ЭЛЕКТРОННОЙ ПОДПИСИ И ДОКУМЕНТОВ, ПОДПИСАННЫХ ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Д.Р. Бабаев*, С.А. Ларинский**
Рязанский государственный радиотехнический университет имени В.Ф. Уткина,
Российская Федерация, Рязань
**babaev.daniil.r@gmail.com*
***s.larinskiy@yandex.ru*

Аннотация. В данной статье описывается основные способы и средства для проверки подлинности документов, подписанных электронной подписью, а также рассматривается проблема подтверждения подлинности документов, заверенных электронной подписью, на бумажных носителях

Ключевые слова: сертификат, электронная подпись, штамп, подтверждение бумажных носителей.

CONFIRMATION OF THE AUTHENTICITY OF AN ELECTRONIC SIGNATURE AND DOCUMENTS, SIGNED WITH AN ELECTRONIC SIGNATURE

D.R. Babaev*, S.A. Larinskiy**
Ryazan State Radio Engineering University named after V.F. Utkin,
Russia, Ryazan
**babaev.daniil.r@gmail.com*
***s.larinskiy@yandex.ru*

The summary. The paper discusses the communication scheme in a MIMO system. Given their main features, advantages and disadvantages, and guidelines for coding information signals. This article describes the main methods and means for verifying the authenticity of documents signed with an electronic signature, and also discusses the problem of confirming the authenticity of documents certified with an electronic signature on paper

Keywords: certificate, electronic signature, stamp, confirmation of paper media..

В современном мире в онлайн режиме заключается огромное количество сделок. Это очень удобно и быстро как для компаний, так и для физических лиц, которые часто совершают покупки в сети интернет. С ростом количества сделок, увеличивается и киберпреступность, поэтому необходимо иметь надежный идентификатор информации в интернете. Такой защитой является электронная подпись.

Согласно действующему законодательству электронной подписью является информация в электронной форме, которая присоединена к другой информации в электронной форме и которая позволяет подтвердить авторство данной электронной информации [1]. Путем электронной подписи устанавливается соответствие между информацией и автором.

В законодательстве, устанавливаются следующие виды электронной подписи:

1) простая электронная подпись;
2) усиленная электронная подпись;
   a) усиленная неквалифицированная электронная подпись;
   b) усиленная квалифицированная электронная подпись.

Также можно выделить виды электронных подписей по связи с документом:

1) присоединенная – электронная подпись вместе с документом объединяются в файл формата «*.sig» (signature), без специальных криптографических программ невозможно открыть и прочесть содержимое подписанного документа (или файла);

2) отсоединенная – электронная подпись формируется в отдельном от документа файле (также формата «*.sig»), содержимое подписанного документа можно посмотреть без специализированного программного обеспечения, для проверки подлинности подписи, соответственно, требуются оба файла;

3) внутри документа – электронная подпись, которая создается в отдельных приложениях («Microsoft Word» или «Acrobat Reader»), обычно генерируется в виде метаданных, и, по сути, является невидимой, однако возможно создание специального поля с подписью.

Существует большое количество способов подделывания электронных подписей. Для противодействуя этому необходимо знать основные способы и средства подтверждения подлинности электронных подписей. Все рассматриваемые средства поддерживают проверку как квалифицированной, так и неквалифицированной электронной подписи. Очевидно, что простую электронную подпись, также проверяют все рассматриваемые сервисы.

«КриптоАРМ ГОСТ»

«КриптоАРМ ГОСТ» (или предыдущая ее версия – «КриптоАРМ») – приложение для создания и проверки электронных подписей, также осуществляющая возможность шифрования данных с использование средств криптографической защиты информации (СКЗИ) «КриптоПро CSP». Приложение сертифицировано ФСБ России и соответствует 63-ФЗ «Об электронной подписи» [2].

Является очень популярным и удобным программным средством проверки документов, подписанных электронной подписью. Может проверять как присоединенную, так и отсоединенную электронную подпись.

При проверке отсоединенной подписи необходимо иметь также и подписанный файл. После проверки есть возможность открыть файл для записи.

Так как в присоединенной подписи файл подписи и документ объединены в один «*.sig» файл, то для проверки необходим всего один файл. После проверки есть возможность извлечения документа из скомпилированного файла подписи для дальнейшего его использования.

Также выходными данными является информация о сертификате, которым был подписан данный документ. Результаты для различных ситуаций приведены на рисунках 1, 2, 3.

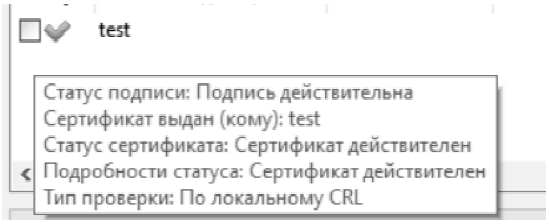


Рис. 1. Результат проверки документа с действительной подписью

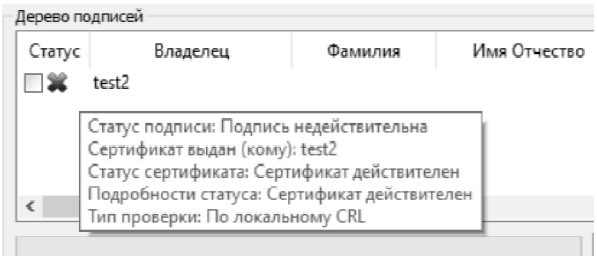


Рис. 2. Результат проверки документа с недействительной подписью

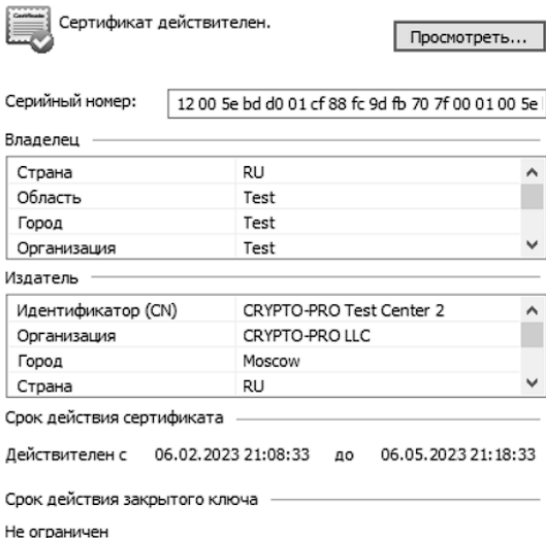


Рис. 3. Сведения о сертификате

Проверка подписи в веб-сервисах

Существуют специализированные веб-сервисы, которые позволяются проверять электронную подпись онлайн. Во многих случаях это оказывается очень удобно.

Одним из таких сервисов является «Контур.Крипто». Это бесплатный сервис от российской компании «СКБ Контур». Лицензирован ФСБ и ФСТЭК России [3].

Сервис позволяет проверить подпись документа и получить информацию о ней, а также информацию о сертификате. Сервис позволяет загрузить файл формата «*.pdf» со всеми выходными данными.

Одной из функциональных особенностей является возможность шифрования и расшифрования документов согласно сертификату получателя.

Минусом данного сервиса является то, что он может проверять электронную подпись только для отсоединенной подписи.

Пример выходных данных представлен на рисунке 4.

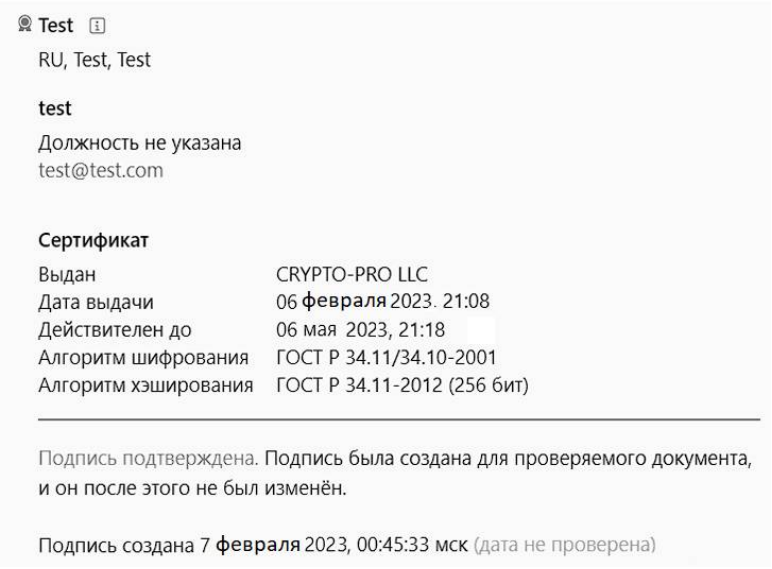


Рис. 4. Результат проверки документа с действующей подписью

Сервис «Контур.Крипто» позволяет проверять не только документы с электронной подписью, но и токены.

Токен – носитель ключей для электронной подписи. Визуально напоминает USB-флеш-накопитель, но в нем встроена защищенная карта памяти с небольшим объемом (до 128 кБайт) относительно обычных USB-флеш-накопителей [4].

Наиболее распространенной в России является продукция компании «Рутокен», который сертифицирован ФСБ и ФСТЭК России. Существует большое разнообразие рутокенов для решения различных задач по идентификации авторов документов. Сервис «Контур.Крипто» поддерживает проверку различных видов рутокенов по их серийным номерам. Сервис находит сертификат подлинности токена, который может быть использован для подтверждения в отделениях некоторых федеральных служб, например, федеральной налоговой службы.

Еще одним возможным веб-сервисом является портал «Госуслуги» [5]. На нем можно проверить действительность присоединенной и отсоединенной подписей и подлинность самого сертификата.

Также реализована возможность проверки подлинности электронной подписи документа по значению хэш-функции. Для проверки вместо электронного документа передается значение его хэш-функции.

Пример выходных данных представлен на рисунке 5.

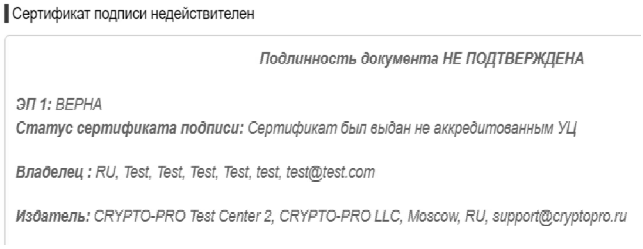


Рис. 5. Результат проверки документа с недействующей подписью

Проверка подписи с помощью плагинов для Word, Excel, PDF

Существует проприетарный плагин «КриптоПро Office Signature», который дает возможность создавать документы, заверенные электронной подписью, а также проверять электронные документы с электронными подписями. Для его функционирования необходимо СКЗИ «КриптоПро CSP».

При успешной проверке документа будет выдана информация о владельце подписи и действительности сертификата, в ином случае будет указана информация о том, что документ содержит недействительную электронную подпись.

Существует плагин также для продуктов Adobe – «КриптоПро PDF». Для его функционирования также необходимо СКЗИ «КриптоПро CSP». Функции аналогичны предыдущему рассмотренному плагину.

Электронная подпись на бумажных документах

В электронном виде электронную подпись можно проверить специальными программами, что невозможно сделать на бумажном варианте документа.

Обычно на распечатках электронная подпись отсутствует вообще, либо выглядит как штамп электронной подписи. Согласно государственным стандартам [6] порядок и составляющие четко установлены. Пример штампа электронной подписи представлен на рисунке 6.

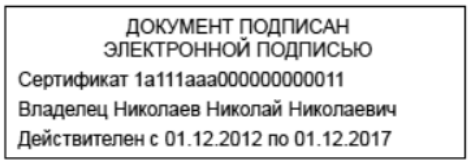


Рис. 6. Пример штампа электронной подписи на бумажном документе

Однако даже такая визуализация не дает документу юридическую силу. Чтобы сделать копию юридически значимой подписи при подаче в суд, федеральную налоговую службу или по требованию других сторон, распечатанный электронный документ следует подписать вручную и поставить на него печать компании и штамп «Копия верна».

Таким образом, все равно необходимо вручную подписывать документ, что роднит его с обычным бумажным документом, чем он, по сути, и является.

Однако даже для штампов электронной подписи (не путать с штампом «Копия верна») нет адекватно работающих универсальных сервисов или программных средств для проверки номеров сертификатов, которые указываются на самом штампе. Авторы статьи считают, что это связано с временной ненадобностью в подтверждении бумажных версий документов, заверенных электронной подписью, так как ситуации, когда это необходимо, не носят массовый характер. Однако все чаще встречаются ситуации, когда необходимо проверить подлинность документа, на котором стоит только штамп электронной подписи без подписей сторон.

Авторы предполагают, что в скором времени появятся подобные сервисы. Однако для их нормальной работы потребуется реестр всех сертификатов для поиска и подтверждения. Встает вопрос о безопасности документов с такими штампами, так как он будет один и его будет очень просто подделать (одна из возможных причин, из-за которой штамп электронной подписи не подтверждает юридическую значимость документа). В таком случае, каждому документу также будет нужен идентификационный номер, который будет связан с номером конкретного сертификата. Соответственно, эти номера можно будет занести в единый реестр документов. Таким образом, получившаяся система обеспечит проверку подлинности конкретного документа от конкретного автора владеющим сертифицированную электронную подпись.

Главная проблема такой системы – необходимы большие вычислительные мощности, которые в данный момент, возможно, применяются на более нужных направлениях. По мнению авторов, в скором времени технологический прогресс преодолеет данный вычислительный барьер, и данная система увидит свет.

Другой же проблемой можно считать то, что в такой системе невозможно обращение с документами, составляющими государственную тайну. Однако ее можно решить разграничением прав доступа к таковым документам или авторам, владеющими соответствующими сертификатами.

Вывод

Все большие обороты в современном мире набирает развитие использования электронного документооборота, и, соответственно, электронных подписей. На данный момент существуют различные системы подтверждения подлинности электронных документов: есть как платные, так и бесплатные утилиты и программные решения. Основной проблемой является вопрос подтверждения подлинности документов, заверенных электронной подписью, на бумажных носителях. Скорее всего, еще не пришло время для решения данного вопроса, но авторы уверены, что в ближайшем будущем найдется способ побороть данную проблему.

Библиографический список

1. Федеральный закон "Об электронной подписи" от 06.04.2011 N 63-ФЗ (последняя редакция) // [Электронный ресурс] – URL: http://www.consultant.ru/document/cons_doc_LAW_112701/ .
2. КриптоАРМ ГОСТ. Сертификаты соответствия // [Электронный ресурс] – URL: https://www.cryptopro.ru/products/partner/crypto-arm.
3. Контур.Безопасность. Лизенции // [Электронный ресурс] – URL: https://kontur.ru/security/licences.
4. Разновидность токенов // [Электронный ресурс] – URL: https://astral.ru/business/usb/.
5. Подтверждение подлинности электронной подписи – Портал государственных услуг Российской Федерации. // [Электронный ресурс] – URL: https://15.gosuslugi.ru/pgu/eds.
6. ГОСТ Р 7.0.97-2016. Национальный стандарт Российской Федерации. Система стандартов по информации, библиотечному и издательскому делу. Организационно-распорядительная документация. Требования к оформлению документов: введ. 08-12-2016 // [Электронный ресурс] – URL: https://docs.cntd.ru/document/1200142871 – Р. 5.

УДК 003.26; ГРНТИ 81.93.29

К ВОПРОСУ О ZK-КРИПТОГРАФИИ И ПРОТОКОЛАХ ДОКАЗАТЕЛЬСТВА С НУЛЕВЫМ РАЗГЛАШЕНИЕМ

А.С. Априщенко*, П.Н. Афонин, Т.И. Калинкина*****
Рязанский государственный радиотехнический университет имени В.Ф. Уткина,
*Российская Федерация, Рязань, *mister.fanat-cska@yandex.ru, **hashaafonin2001@mail.ru, ***tkalinkina2014@mail.ru.*

Аннотация. В работе рассматриваются протоколы с нулевым разглашением. Приводятся их основные особенности, достоинства и недостатки, а также примеры использования их в настоящее время.
Ключевые слова: Zero Knowledge (ZK), протокол, криптография, доказательство, аутентификации, криптопротокол, информация, zk-SNARK, zk-STARK.

ON THE ISSUE OF ZK CRYPTOGRAPHY AND ZERO-KNOWLEDGE PROOF PROTOCOLS

A.S. Aprishchenko*, P.N. Afonin, T.I. Kalinkina*****
Ryazan State Radio Engineering University named after V.F. Utkin,
*Russian Federation, Ryazan, *mister.fanat-cska@yandex.ru, **hashaafonin2001@mail.ru, ***tkalinkina2014@mail.ru.*

Annotation. The paper considers protocols with zero disclosure. Their main features, advantages and disadvantages are given, as well as examples of their use at the present time.
Keywords: Zero Knowledge (ZK), Protocol, Cryptography, Proof, authentication, cryptoprotocol Information, zk-SNARK, zk-STARK.

Существует три способа аутентификации – что субъект знает, что имеет и кто субъект есть. Проще всего реализовать первый способ, который определяется знанием какого-то секрета, например, пароля. Системы такого типа являются самыми распространенными в настоящее время, поскольку их проще всего создавать. Сейчас это самый широко используемый способ аутентификации – вход в любую систему при помощи пароля. Но есть у этого способа и существенный недостаток. Безопасность зависит только от держателя этого секрета или пароля, который полагается на безопасное хранение паролей на серверах. Зачастую люди придумывают очень простые пароли, чтобы упростить себе жизнь, так как вход в систему при помощи пароля подразумевается у всех сервисов, используемых человеком в повседневной жизни. Поэтому существует другой подход реализации аутентификации.

Простыми словами аутентификация – это доказательство. Возьмем для примера мальчика Гришу и девочку Машу. Гриша должен убедиться, что Маша точно знает секрет. Маша

в свою очередь не желает распространять свой секрет, но должна доказать Грише, что она точно его знает, причем Гриша поверит Маше. Кроме того, Маша точно не желает, чтобы Гриша воспользовался секретом и выдал себя за неё. Такую схему реализуют протоколы с нулевым разглашением. То есть Гриша задаёт вопросы Маше, на которые ответ есть только у неё, но проверить правильность её ответа могут многие. Скорее всего, вероятность обмана крайне мала. У Гриши не получится воспользоваться доказательствами Маши, чтобы себя выдать за неё. Это и есть нулевое разглашение.

Для определенности введем в статье следующее определение ZK-криптографии.

Будем понимать под ZK-криптографией раздел криптографии, изучающий вопросы, посвященные криптопротоколам zero-knowledge proof, т.е. протоколам доказательства с нулевым разглашением.

Сокращение ZK - это аббревиатура от английских слов Zero Knowledge.

Основополагающим свойством подобных протоколов является нулевое разглашение – это свойство, обеспечивающее такое выполнение протокола доказательства, что информация об утверждении, которое подлежит доказательству, кроме факта его истинности, не может быть получена лицом, без прав доступа к ней.

Различают следующие протоколы доказательства с нулевым разглашением [2]:

- интерактивное/неинтерактивное доказательство;
- протокол доказательства знания (некоторого факта);
- протокол доказательства умения решать некоторую задачу.

Рассмотрим детальнее реализацию некоторые из этих протоколов. Интерактивное доказательство может выполняться минимум двумя участниками. Один из них является доказывающим, а второй - проверяющим. В процессе взаимодействия между собой участники обмениваются сообщениями, которые зависят от случайных чисел, содержащихся в секрете [2]. Цель доказывающего - убедить проверяющего в истинности некоторого утверждения. Проверяющий либо считает доказательство истинным, либо - ложным. Понятие «доказательство» здесь носит вероятностный характер и определяется двумя вероятностями. Если доказываемое утверждение истинно, то доказательство считается верным с вероятностью, стремящейся к единице при увеличении количества повторений протокола. Если доказываемое утверждение ложно, то вероятность правильности доказательства будет стремиться к нулю при увеличении количества повторений протокола.

Доказательство будет интерактивным, если проверяющий непосредственно в реальном времени опрашивает доказывающего. Данный вид доказательства является более трудоемким, если требуется доказать утверждение нескольким проверяющим. В этом случае доказывающему придется повторять доказательство для каждого проверяющего.

Доказательство будет неинтерактивным, если между доказывающим и проверяющим нет непосредственного взаимодействия. Допустим, что сначала доказывающий может сделать какое – либо утверждение, а затем проверяющий убеждается в его подлинности.

Доказательство неинтерактивное представляет собой лучший способ доказать утверждение более чем одному проверяющему, при этом не увеличивая требуемые ресурсы и затраты.

Протокол доказательства знания представляет собой итеративный процесс верификации, в котором убеждающий доказывает повторно верифицирующего в том, что у него есть совершенно секретная информация, не раскрывая ее.

Рассмотрим конкретные примеры реализации таких протоколов.

Пример 1.

На простом примере покажем, как можно доказать что-то, при этом не давая конкретную информацию о том, что собираемся доказать.

Пусть девочка Маша и мальчик Гриша собрали какое-то количество монет (номинал монет не имеет значения) и хотят убедиться, одинаковое у них количество этих самых монет или разное.

Гриша берет четыре шкатулки и пишет цифры на каждой из них: «10» на первой, «20» на второй, «30» на третьей и «40» на четвертой соответственно (см. рисунок 1).

Рис. 1. Четыре шкатулки

Он собрал 20 монет, поэтому оставляет себе ключ от шкатулки с цифрой 20, а остальные ключи от оставшихся шкатулок выбрасывает, у него их больше нет.

После этого Маша берет 4 листа бумаги и рисует на одном из них плюс, а на остальных листах минусы. Потом сворачивает их и засовывает эти бумажки в щёлки на шкатулках (см. рисунок 2).

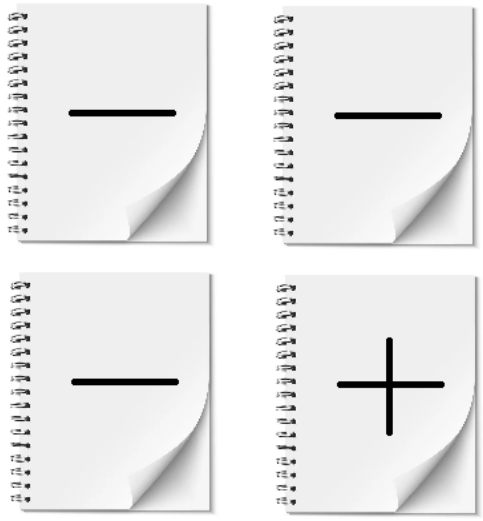


Рис. 2. Листы бумаги

После чего приходит Гриша в комнату, открывает единственным ключом свою шкатулку и видит там записку от Маши, а именно листок со знаком минус. То есть теперь он знает, что Маша собрала не 20 монет, сколько она собрала – он не знает. Девочка не знает сколько собрал мальчик. Никто ничего не знает. Но, однако, ответ на свой вопрос они получили: они убедились в том, что собрали разное число монет.

Пример 2.

Представим коридор, который ведёт в кольцевую комнату, то есть комнату в виде окружности с одним входом и дверью, разделяющую комнату на две части. Чтобы пройти через дверь, нужно ввести секретный пароль. Итак, возьмем всё тех же Машу и Гришу. Допустим, Маша хочет доказать Грише, что она знает пароль, но при этом оставит его в тайне. Пока Гриша находится в точке А, Маша идет к двери. А Гриша их точки А перемещается в точку Б и оттуда просит Машу выйти, например, по пути 2. В данном случае, чтобы Маше выйти по этому пути, она должна открыть дверь. Если она знает пароль, то ей это удастся и таким образом, не называя пароля докажет Грише, что она этот пароль знает. Если же она не сможет пройти путем 2, то это докажет, что Маша пароль не знает от этой двери. Однако возможна другая ситуация, когда Гриша попросит выйти Машу тем же путем, которым она вошла в эту комнату, тогда ей не потребуется открывать дверь и она не сможет подтвердить доказать свое знание пароля от двери. Получается, что каждый раз вероятность незнания пароля Маши равна 50%. Проделав же этот эксперимент много раз Гриша точно убедится в том, что Маша открывала дверь при помощи пароля, так как всегда угадывать путь по которому Маша вошла в комнату невозможно. Гриша соглашается остаться перед комнатой в коридоре и не видит в какую сторону пойдет Маша (допустим, путь 1) (см. рисунок 3).

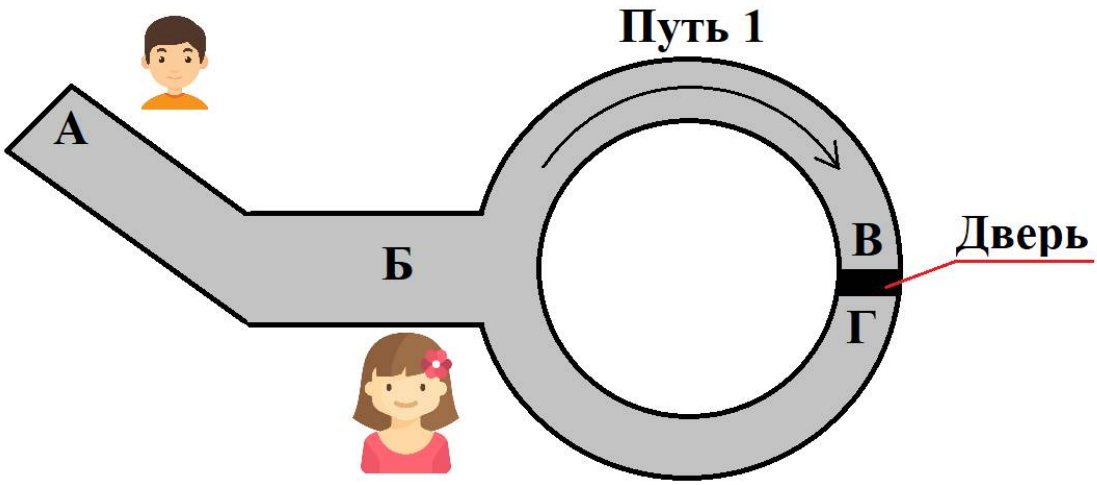


Рис. 3. Экспериментальная комната, путь 1

Вскоре Гриша говорит, чтобы Маша появилась с одной из сторон комнаты (возьмем путь номер 2) (см. рисунок 4).

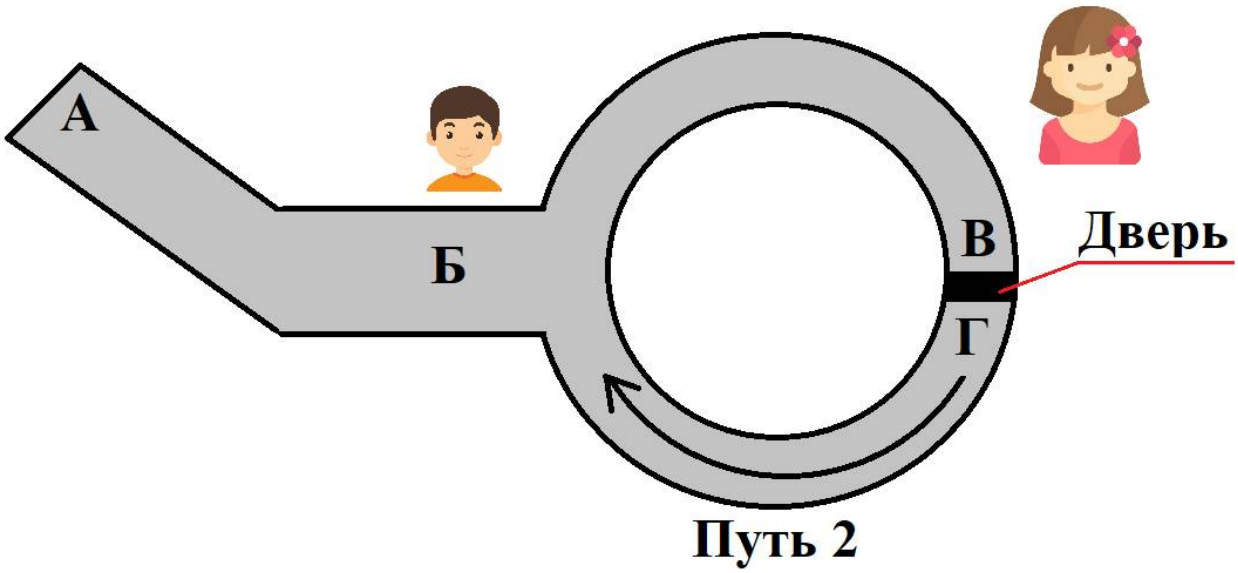


Рис. 4. Экспериментальная комната, путь 2

Если Маша действительно знает пароль и ее начальный путь не совпадает с Гришиным, она откроет дверь и точно окажется на выбранном им пути. В противном случае Маша могла пойти именно по тому пути, который выбрал Гриша и ей бы не пришлось открывать дверь. Проделав же этот эксперимент много раз Гриша точно убедится в том, что Маша открывала дверь при помощи пароля, так как всегда угадывать путь невозможно.

Такой пример также показывает концепцию доказательств с нулевым разглашением. Они могут применяться для подтверждения владения определенной информацией, не раскрывая ее саму.

Что такое zk-SNARK и zk-STARK?

Протоколы доказательства с нулевым разглашением можно разделить на две группы. Различаются они наличием или отсутствием фазы доверенной настройки (установки), с английского языка это переводится как Trusted Setup Phase. Рассмотрим на примере фазу доверенной настройки. Для этого проанализируем первый из приведенных в статье примеров реализации протокола доказательства с нулевым разглашением. Для того чтобы убедиться одинаковое или разное количество монет собрано, девочка и мальчик выполняют подготовительные действия. Так, Гриша берет 4 шкатулки и пишет на них соответствующие цифры. Затем в шкатулку с цифрой 20 кладет свои монеты. Маша, в свою очередь, берет 4 листа бумаги и рисует на одном знак плюс, а на остальных трех знак минус. Эти листы бумаги она сворачивает и помещает их в шкатулки. Все действия Маши и Гриши, как видно, относятся к действиям, подготавливающим получение ответа на интересующий их вопрос.

Кроме фазы доверенной настройки протоколы доказательства с нулевым разглашением характеризуются двумя аргументами: zk-SNARK и zk -STARK.

Механизм написания на бумажке и закрытия её в шкатулке ключом является примером zk-SNARK (пример 1).

Расшифровывается это так:

- zk - Zero Knowledge
- S - Succinct (быстрый, скорый)
- N – Non-interactive (не интерактивный)
- AR – Arguments (набор)
- K – Knowledge (знаний)

zk-STARK:

• T – transparent (понятный, явный)

zk-STARK более современная технология, нежели zk-SNARK, она появилась на 5 лет позже.

В [4] следующим образом поясняются эти аргументы:

«zk-SNARK или же краткий неинтерактивный аргумент знания с нулевым разглашением — это решение, в котором:

- краткий означает, что размер доказательства должен быть достаточно маленьким, чтобы его можно было проверить за короткое время;

- неинтерактивный указывает на то, что это конструкция, где между проверяющим и доказывающим нет двусторонней связи».

Такой аргумент создаёт модель, когда доказывающий может убедить проверяющего в том, что существует некоторый факт или информация, при этом не говоря о том, что из себя он или она представляет.

Нулевое разглашение в свою очередь говорит о том, что если утверждение истинно, то в этом случае проверяющий никак не выяснит ровным счетом ничего, кроме того факта, что оно на самом деле истинно.

zk-STARK или же масштабируемый прозрачный аргумент знания с нулевым разглашением это усовершенствованная версия zk-SNARK.

Главная разница между ними заключается в том, что zk-SNARK нуждаются в доверенной фазе настройки, которая нужна для создания доказательств с нулевым разглашением. Если информация попадет в чужие руки, то сторонние участники смогут подделать данные доказательства. В свою очередь zk-STARK используют симметричное шифрование, чтобы такая информация не стала публичной.

zk -STARK ликвидирует основной недостаток zk-SNARK: его необходимость в доверенной настройки. Для zk-SNARK такое условие является необходимым, а для zk -STARK это не нужно. Он следует более простым криптографическим предположениям.

zk -STARK использует публично проверяемую случайность. Такие системы создают доверие, в котором может убедиться любой проверяющий. Из этого и следует буква "T" в названии, означающая "Transparent": прозрачный.

Заключение

На основе протоколов доказательство с нулевым разглашением имеет свои достоинства и недостатки. В качестве достоинств можно выделить следующее:

1. Повышение приватности пользователей в различных сетях.
2. Усиление информационной безопасности за счет использования более эффективных способов проверки и верификации взамен не эффективных.
3. Повышение пропускной способности и улучшение масштабируемости.

К недостаткам можно отнести:

1. Требование значительных вычислительных мощностей для интерактивных протоколов.
2. Возможность обмана в фазе доверенной настройки в случае использования zk-SNARK.
3. В том случае, если инициатор транзакции забудет свой исходный код доступа, все данные будут утеряны навсегда.

В настоящее время протоколы доказательства с нулевым разглашением используются государственными органами для определения происхождения определенных данных без необходимости доказывать, где и от кого они получили информацию.

Есть несколько примеров платформ самоидентификации, которые позволяют третьим сторонам, таким как правоохранительные органы, определять, есть ли у человека действи-

тельные водительские права, без необходимости предоставления человеком чего-либо, кроме своего идентификационного номера.

Библиографический список

1. Молдовян А.А., Молдовян Д.Н., Левина А.Б. ПРОТОКОЛЫ АУТЕНТИФИКАЦИИ С НУЛЕВЫМ РАЗГЛАШЕНИЕМ СЕКРЕТА / - М: СПБ, 2016.

2.Черемушкин А.В. Криптографические протоколы. Основные свойства и уязвимости/ - М: Москва, 2009.

3. Hannu A. Aronsson. Zero Knowledge Protocols and Small Systems. Department of Computer Science, Helsinki University of Technology, 2007.

4. Zero-Knowledge Proof — Как работает криптография с нулевым разглашением - Простыми словами, // [Электронный ресурс].- 2023.- Режим доступа: https://vc.ru/s/1355052-dnevnik-neydachnika/603550-zero-knowledge-proof-kak-rabotaet-kriptografiya-s-nulevym-razglasheniem-prostymi-slovami - Дата доступа: 09.02.2023.

УДК 004.056.53; ГРНТИ 81.93.29

АКТУАЛЬНЫЕ УЯЗВИМОСТИ СЕТЕВЫХ ПРИЛОЖЕНИЙ НА ПРИМЕРЕ OWASP TOP TEN

О.В. Сидоров, Ю.М. Кузьмин

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, bobrjax@gmail.com, kuzmin_yurii@mail.ru

Аннотация. В работе рассматриваются актуальные уязвимости веб приложений, представляющие наибольшую опасность для организаций и объясняются основные механизмы их предотвращения.

Ключевые слова: веб приложения, уязвимости, OWASP Top Ten, инъекция, доступ, угрозы.

ACTUAL WEB APPLICATIONS VULNERABILITIES ACCORDING TO OWASP TOP TEN

O.V. Sidorov, U.M. Kuzmin

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, bobrjax@gmail.com, kuzmin_yurii@mail.ru

The summary. Actual vulnerabilities of web applications showing the highest level of danger for companies are discussed in the paper. The main ways of preventing them are shown.

Keywords: web applications, vulnerabilities, OWASP Top Ten, injection, access, threats.

В современном мире подавляющее число атак с помощью вредоносного программного обеспечения приходится на сайты и веб-приложения. Такое положение дел провоцирует разработчиков уделять больше внимания безопасности проектируемых ими приложений, что позволяет значительно сократить или даже свести к нулю ущерб, получаемый от атак на сетевую инфраструктуру компании. Такой подход позволяет уменьшить вероятность репутационных и других потерь.

Реализация мер безопасности на этом этапе сводится к анализу потенциальных угроз для веб-приложения и принятия мер к их предотвращению. Почерпнуть необходимую информацию, в частности, можно из списка OWASP TOP 10[1].

OWASP Top Ten (открытый проект обеспечения безопасности веб приложений) — это открытый отчёт, перечисляющий основные угрозы безопасности сетевых приложений. Отчёт постоянно актуализируется, приводя 10 наиболее серьёзных рисков, возникающих в организациях, использующих сетевые приложения. Для минимизации рисков и смягчения ущерба всем организациям рекомендуется учитывать данные отчёта при выстраивании корпоративных процессов.

Этот документ формируется на основе мнений специалистов по информационной безопасности всего мира. Угрозы в отчёте ранжированы на основе частоты встречаемости угрозы и её потенциального ущерба, что позволяет компаниям грамотно распределять усилия по обеспечению безопасности сетевой инфраструктуры.

В последнем отчете (2021) OWASP перечислены 10 основных уязвимостей:

1. Нарушенный контроль доступа (Broken Access control).
2. Ошибки криптографии (Cryptographic Failures).
3. Инъекции (Injections).
4. Небезопасное проектирование (Insecure Design).
5. Некорректная настройка параметров безопасности (Security Misconfiguration).
6. Уязвимые и устаревшие компоненты (Vulnerable and Outdated Components).
7. Ошибки идентификации и аутентификации (Identification and Authentication Failures).
8. Нарушения целостности ПО и данных (Software and Data Integrity Failures).
9. Нарушение безопасности журналирования и мониторинга.
10. Подделка запроса на стороне сервера.

Нарушенный контроль доступа

Грамотно выстроенное управление контролем доступа позволяет избежать действий пользователя за пределами полномочий, которыми он наделен. Ошибки настройки политики доступа обычно ведут к обходу злоумышленником заданных ограничений и получению несанкционированного доступа к системам и конфиденциальным данным, а также потенциальным вектором развития атаки может стать эскалация привилегий.

Основным механизмом предотвращения этой угрозы является следование политике «всё запрещено». При распределении прав в соответствии с этой политикой риск наделения пользователей излишними правами минимален. Так же для ограничения доступа к странице администрирования сервиса не должен использоваться метод безопасности за счёт неизвестности.

Ошибки криптографии

Конфиденциальная информация, передаваемая по техническим каналам связи, должна быть надёжно защищена. В подавляющем большинстве случаев для такой защиты используются криптографические методы защиты информации. Криптографические уязвимости возникают, например, в результате недостаточной криптостойкости используемого алгоритма шифрования, недостаточного уровня энтропии генератора случайных чисел, применяемого для нужд криптографического алгоритма, использование открытых протоколов передачи информации в сети.

Для предотвращения основных путей эксплуатации криптографических уязвимостей необходимо:

1. Использовать криптографические алгоритмы достаточной криптостойкости.
2. Использовать для передачи конфиденциальной информации в сети только протоколы, предусматривающие шифрование.
3. Не хранить конфиденциальную информацию без необходимости.
4. Отключить кеширование конфиденциальной информации.
5. Следовать рекомендациям регуляторов по обращению с конфиденциальной информацией.

Инъекции

Этот тип уязвимости возникает при использовании нарушителем ненадёжных данных в запросы или формы. Таким образом командный интерпретатор со стороны сервера может воспринять такие данные как команду, тем самым нарушитель может удалённо исполнять команды (remote code execution) или получить доступ к конфиденциальным данным, например, вставив в неэкранированную форму код SQL-запроса.

Для предотвращения эксплуатации инъекций при написании серверной части приложения необходимо экранировать все элементы кода и переменные, значение которых передаётся формами и полями, предполагающие ввод данных пользователем, а также проверять все вводимые пользователем данные. Помимо этого, для исключения возможности проведения SQL инъекции можно использовать СУБД типа NoSQL.

Небезопасное проектирование

Небезопасное проектирование является широкой категорией различны уязвимостей, возникающих при игнорировании общепринятых методологий построения архитектуры приложения, начиная с фазы планирования непосредственно перед введением продукта в использование. Следует заметить, что небезопасное проектирование отличается от небезопасного встраивания, и даже проведённое почти идеально встраивание не предотвращает уязвимости, возникающие при небезопасном проектировании. Такие уязвимости возникают, когда разработчикам, тестировщикам и команде безопасности не удаётся обнаружить и оценить уязвимости во время фазы построения архитектуры приложения.

Одним из основных методов исключения этой категории уязвимостей является использование подхода тестирования Shift Left, который заключается в привлечении команды тестирования непосредственно на первых фаза разработки ПО, в отличие от общепринятого подхода тестирования как одного из поздних этапов цикла разработки.

Некорректная настройка параметров безопасности

Уязвимости этого типа появляются при неправильном конфигурировании приложений и отдельных компонентов, серверов. Основной механизм возникновения этих уязвимостей — это нежелание системного администратора менять конфигурацию по умолчанию, которая, как известно, далеко не всегда обеспечивает наилучшие параметры безопасности используемого компонента или сервиса, а также недостаточная квалификация системного администратора в вопросах информационной безопасности.

Методами предотвращения угроз этого типа являются:

1. Тщательное конфигурирование используемых компонентов и сервисов.
2. Отключение компонентов, удаление кусков кода не являющихся необходимыми во избежание недекларированных возможностей.
3. Смена логинов и паролей по умолчанию.

Уязвимые и устаревшие компоненты

Риски возникновения этой уязвимостей появляются при использовании сторонних компонентов как на клиентской, так и на серверной стороне. Зачастую бывает, что такие компоненты не тестируются должным образом и, как следствие, могут иметь незакрытые бреши в безопасности, успешно эксплуатируемые злоумышленниками. Также эта уязвимость возникает при пренебрежении своевременным обновлением используемых компонентов.

Для того, чтобы предотвратить данный тип уязвимостей достаточно вовремя обновлять используемые компоненты, по возможности сократить использование open source ком-

понентов вследствие потенциального наличия уязвимостей, а также загрузки необходимых компонентов только с официальных источников.

Ошибки идентификации и аутентификации

Ошибки идентификации и аутентификации могут возникать, когда функции, относящиеся к определению пользователя, аутентификации и управлению сессией внедрены неверно или недостаточно защищены приложением. Эксплуатация данной уязвимости проводится такими техниками, как грубый перебор (brute force), перехват cookies, «фиксация» сессии. Обычно приводит к компрометации паролей, ключей, токенов сессии.

Использование таких методов, как многофакторная аутентификация, ограничение количества запросов в секунду с одного IP адреса, капча, ужесточение парольной политики, а также защищённые менеджеры сессий позволяет успешно противостоять основным уязвимостям данного типа.

Нарушения целостности ПО и данных [2]

Современные методологии разработки ПО нередко вынуждают разработчиков использовать плагины, модули и библиотеки из общедоступных репозиториев, ненадежных источников. Нарушения целостности программного обеспечения и данных возникают, когда критически важные данные и обновления программного обеспечения загружаются без проверки их целостности. При отсутствии надлежащей проверки нарушения в целостности программного обеспечения и данных делают приложения уязвимыми для несанкционированного раскрытия информации, компрометации системы или внедрения вредоносного кода.

Для устранения данной уязвимости используются такие методы, как сверка хэш-сумм загруженных компонентов с таковыми, предоставляемыми разработчиками на официальном источнике, использования цифровой подписи для подтверждения целостности, загрузка компонентов и обновлений только из доверенных источников, а также проверку исходных кодов приложений на этапах разработки.

Нарушение безопасности журналирования и мониторинга

Журналирование и мониторинг играют крайне важную роль в обнаружении нарушений и реагировании на них. Нарушения не могут быть обнаружены без ведения журнала и мониторинга. Система уязвима, когда события, подлежащие аудиту, такие как неудачные попытки входа, ошибки и предупреждения и иные значимые системные события не фиксируются, файлы логов хранятся локально, логи не анализируются на подозрительную активность.

Во избежание эксплуатации данной уязвимости необходимо удостовериться, что все значимые события в системе фиксируются и сообщение о каждом событии содержит минимальный набор сведений о нём. Используются также постоянный мониторинг и средства анализа логов на подозрительную активность.

Подделка запроса на стороне сервера

Подделка запросов на стороне сервера - это уязвимость в системе безопасности, которая позволяет злоумышленнику отправлять запросы от имени скомпрометированного сервера. Например, риск атаки возникает, если приложение использует непроверенные внешние данные для генерации запросов. Злоумышленник может спровоцировать отправку вредоносных запросов к ресурсам, которые не доступны ему напрямую, но доступны серверу.

Для минимизации угроз данного типа формируются списки разрешённых URL, отключается перенаправление HTTP, встраиваются системы валидации ответов сервера.

Таким образом, анализ современного список уязвимостей позволяет минимизировать и избежать наиболее активно эксплуатируемые угрозы сетевым приложениям, а также определить направления развития существующей системы безопасности приложения.

Библиографический список

1. OWASP Top Ten. [Электронный ресурс]. - Режим доступа: https://owasp.org/Top10/. - Дата доступа: 18.02.2023.
2. Хоффман Э. Безопасность веб-приложений. – М.: Пресс книга. - 2022. – 336 с.

ИСПОЛЬЗОВАНИЕ СТЕГАНОГРАФИЧЕСКИХ МЕТОДОВ ДЛЯ ПОДТВЕРЖДЕНИЯ ПОДЛИННОСТИ ДАННЫХ

М.И. Колесников
Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, makkoar@gmail.com

Аннотация. В работе рассматриваются способы использования стеганографических методов для подтверждения подлинности данных. Приводятся их основные особенности, достоинства и недостатки, а также сами стеганографические методы.
Ключевые слова: наименее значимый бит (LSB), разность значений пикселей (PVD), изменение уровня серого (GLM), стеганография, передача данных, данные, изображение.

USING STEGANOGRAPHIC METHODS FOR DATA AUTHENTICATION

M.I. Kolesnikov
Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, makkoar@gmail.com

Annotation. The paper discusses ways to use shorthand methods to verify the authenticity of data. Their main features, advantages and disadvantages, as well as the steganographic methods themselves are given.
Keywords: least significant bit (LSB), pixel value difference (PVD), gray level modification (GLM), steganography, data transmission, data, image.

Одной из важных проблем в современности является трудность в проверки подлинности данных. Порой существует необходимость в проверке данных на подлинность, например является ли какая-либо статья или изображение оригинальным или же оно было видоизменено третьими лицами. Существует ряд способов для осуществления проверки подлинности данных, но у всех есть свои достоинства и недостатки. В данной статье я собираюсь рассмотреть один из таких способов, которым является стеганография.

Изначально, стеганография появилась для того, чтобы скрытно передавать информацию (сообщения) вместе с другой информацией, например для передачи зашифрованного в тексте сообщения, которое сможет получить и расшифровать получатель этого сообщения. То-есть по факту адресат получает два сообщения, одно из которых является заглушкой, которая не привлекает внимание, а другое - зашифрованное сообщение, которое имеет некоторую ценность для получателя.

В современности стеганографические методы плотно связанны с криптографией и используют её в своём составе и в отличии от самой криптографии стеганография скрывает не само содержимое сообщения, а сам факт существования этого сообщения**Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**.

Стеганография имеет ряд преимуществ над чистой криптографией, например замаскированные с помощью неё сообщения не привлекают к себе внимания или она позволяет

проверить подлинность присланного сообщения. Недостатком же стеганографии является очень малый объём передачи скрытой информации, что на самом деле довольно ситуативно, например при использовании в качестве стеганографического сообщения изображения, можно передать на порядок больше данных, чем при сокрытии информации в обычном тексте, а при сокрытии информации в звуковом файле и того больше.

В данной статье рассматриваются различные стеганографические классы, методы (алгоритмы) и потенциальные способы применения некоторых из них.

Классы стеганографии

Всего выделено 4 основных класса стеганографии **Ошибка! Источник ссылки не найден.**:

- классическая;
- компьютерная;
- цифровая;
- сетевая.

Вкратце о каждом:

Классическая стеганография

Классическая стеганография заключается в использовании физических объектах, в которых можно скрыть информацию. Чаще всего такими объектами является сообщения на бумажных носителях. Например, можно скрыть сообщение в тексте с помощью заранее оговоренных специальных символах, о которых будут знать только отправитель и получатель, или с помощью специальных чернил, которые будут незаметно выделять нужные буквы скрытого сообщения.

Компьютерная стеганография

Компьютерная стеганография заключается в использовании возможностей компьютерных платформ, например использовании неиспользуемых или редко используемых областях форматов файлов или подмене символов в названии файлов**Ошибка! Источник ссылки не найден.**.

Цифровая стеганография

Цифровая стеганография является частичным ответвлением от классической, так как часто использует похожие методы для своей реализации. В большей части, отличие данного направления от классического заключается в том, что данные скрываются не на физических носителях, а в цифровых объектах. Примером является использование символов в тексте, внешне похожих на оригинальный символ, например О (буква «О») или 0 (нуль), или символы пробела и неразрывного пробела, которые могут служить маркерами для получателя. Другим примером можно назвать сокрытие информации в изображении, посредством подмены определённых пикселей так, чтобы исходное изображение было не сильно искажено, например можно подменить ряд пикселей на почти чёрный (R0, G0, B1, A255), совокупность которых может содержать в себе текст, закодированный бинарным кодом (места изменяемых пикселей так же должна быть заранее оговорена с получателем сообщения)**Ошибка! Источник ссылки не найден.**.

Сетевая стеганография

Сетевая стеганография заключается в передаче информации посредством использования специальных сетевых протоколов передачи данных. В частности, выделяют два её вида, такие как WLAN-стеганография (передача сообщения по беспроводной сети) и LACK-стеганография (передача сообщения с использованием IP телефонии)**Ошибка! Источник ссылки не найден.**.

Методы стеганографии

Теперь перейдём к рассмотрению самих методов стеганографии.

Существует довольно разных стеганографических алгоритмов, потому все их рассматривать нет смысла. Здесь мы рассматриваем лишь некоторые, которые являются довольно перспективными.

Одним из таких методов является метод «вливания» скрытой информации. В случае с текстом, изображением или звуком, происходит наложение скрытого изображения на текст, изображения или звука поверх оригинала. В случае с текстом, всё немного сложнее и менее перспективно, так как труднее «влить» большой объём скрытой информации, так как чем больше скрытого текста будет в него вставлено, тем больше вероятность, что возникнут какие-либо подозрения у стороннего наблюдателя и скрытое послание будет обнаружено. В случае с изображением или же звуком объём скрытого сообщения почти не повлияет на оригинальное изображение и звук и повышается шанс, что сообщение не будет обнаружено сторонним наблюдателем и будет доставлено не перехваченным.

На практике стеганография позволяет проверить является-ли сообщение подлинным или же было перехвачено и видоизменено, например отправитель встроил в текст/изображение/звук свой уникальную скрытую информацию, по которой получатель сможет проверить, что сообщение не было изменено. В случае если сообщение было перехвачено и изменено, то из-за незнания, злоумышленник с высокой долей вероятности изменит и уникальную информацию отправителя, которая была встроена в сообщение и при его доставке получателю, он сможет проверить было-ли изменено сообщение и заметить несостыковки с уникальной информацией, которая по какой-то причине была видоизменена. По факту, данный способ использования стеганографии можно сравнить с использованием водяных знаков, но в отличии оригинальных водяных знаках об их присутствии знает только получатель и отправитель.

Рассмотрим некоторые способы сокрытия информации в изображении:

Наименее значимый бит (LSB)

Этот способ заключается в выделении наименее значимых бит в изображении, которые будут заменены на скрытое сообщение. Поскольку изменяются лишь наименее значимые биты то разница между оригиналом и изменённым изображением почти незаметна. Данный метод используется в форматах изображений, которые не используют сжатие или со сжатием без потерь.

Разность значений пикселей (PVD)

Этот способ учитывает то, что на участках, где значение яркости меняется незначительно изменение будет более заметно, чем на участках, содержащих более значительные перепады яркости и информация записывается именно на таких участках, где перепады яркости более выражены.

Изменение уровня серого (GLM)

Этот способ заключается в том, что изменяется чётность значения яркости изображения в чёрно-белом представлении, тоесть в каждый пиксель встраивается 1 бит скрытой информации.

Эффективность и стойкость стеганографических алгоритмов к раскрытию скрытой информации

Стеганографические методы имеют довольно высокую эффективность, так как их довольно затруднительно отследить, но если скрытое сообщение обнаружат и смогут выяснить, какой стеганографический алгоритм использовался при сокрытии информации, то стойкость этого метода оставит желать лучшего. Однако есть некоторые способы, которые позволяют значительно поднять стойкость и эффективность стеганографических алгоритмов. Примером

такого способа является совмещение в себе стеганографических методов сокрытия информации и методов шифрования. При таком подходе эффективность и стойкость скрытой информации начинает напрямую зависеть от применяемого метода шифрования. (Чем выше эффективность и криптостойкость алгоритмов шифрования, тем выше эффективность и стойкость стеганографического алгоритма)

Для оценки стеганографических методов нужна система показателей оценки их качества. Показатели качества стеганографических алгоритмов можно выразить с помощью показателей, характеризующих стеганографический метод с позиции скрытности, т.е. стойкости метода к выявлению факта наличия скрытого сообщения в изображении.

Качественно скрытность стеганографического алгоритма может быть определена при помощи экспертных оценок. При наличии исходного изображения стойкость стеганографического метода может быть оценена с помощью количественных разностных и корреляционных показателей. Наиболее широко используемым показателем является «пиковое отношение сигнал-шум» (ПОСШ). Данная величина показывает степень отличия исходного изображения от изменённого и измеряется в децибелах (дБ). ПОСШ вычисляется на основе следующей формулы:

$$\text{ПОСШ} = \frac{m*n*max(I_{i,j})^2}{\sum_{i,j}(I_{i,j}-C_{i,j})^2},$$

где m*n - размер изображения;

$I_{i,j}$ - пиксель исходного изображения;

$C_{i,j}$ - пиксель изменённого изображения**Ошибка! Источник ссылки не найден.**.

Средства для сокрытия информации и анализа на наличие скрытой информации

Встроить данные в изображение можно разными способами в том числе и вручную, через любой графический редактор, но такой метод является очень сложным и неудобным в отличии от способа встраивать данные с помощью специальных программ. Примером таких программ может послужить:

- «OutGuess», «JSTEG» и «JPHS» - позволяют скрывать данные в JPEG изображениях (Данные, спрятанные с помощью программ «JSTEG» и «JPHS» являются довольно не стойкими, так как есть возможность пассивного (автоматического детектирования наличия скрытых данных));
- «Gifshuffle» и «Hide-and-Seek» - позволяют скрывать данные в GIF файлах («Gifshuffle» имеет довольно малый объём скрываемых данных вне зависимости от размера контейнера (GIF-изображения));
- «Steganos» - позволяет скрывать данные в графических файлах BMP, DIB, VOC, WAV, ASCII;
- «Steghide» - позволяет скрывать данные в графических BMP- и звуковых WAV- и AU- файлах;
- «DC-Stegano» - позволяет скрывать данные в графических файлах в формате PCX;
- «Invisible Secrets 2002», «Steganos for Windows» и «Hide4PGP» - позволяют скрывать данные в аудиоданные в формате PCM, WAV, BMP, DIB, VOC, HTML, ASCII;
- «StegoWav» - позволяет скрывать данные в аудиоданные в формате PCM (WAV);
- «Steaghan» - позволяет скрывать данные в графических BMP- и звуковых WAV-файлах;

- «S-Tools» - позволяет скрывать данные в аудиоданных в формате PCM (WAV);
- «MP3Stego» и «UnderMP3 Cover» - позволяют скрывать данные в аудиофайлах в формате MP3 (Имеют низкую стойкость к пассивному стегоанализу);
- «WNS (Белый шумовой шторм)» - универсальная программа стеганографии, которая может скрывать информацию в большинстве известных форматов;
- «Covert_TCP» - программа управляет TCP_IP заголовком и передает с каждым файлом один скрытый байт;
- «SecurEngine» - позволяет скрывать данные в текстовые файлы, графические файлы (BMP, JPG) и аудиофайлы (WAV).

Большинство вышеуказанных программ при сокрытии информации используют дополнительное шифрование. Так же многие из этих программ умеют не только скрывать информацию, но и обнаруживать, при наличии у получателя информации специального ключа (некоторые программы не требуют ключа при сокрытии и раскрытии информации, что вредит стойкости используемого метода).

Выводы

В качестве заключения данной статьи можно сделать некоторые выводы. В статье проводится классификация стеганографии и описание каждого класса, особое внимание было уделено цифровой стеганографии, рассказано о некоторых существующих методах сокрытия информации и способы применения этих методов на практике. Так же был раскрыт способ выяснить эффективность и стойкость в виде оценки показателя скрытности стеганографических алгоритмов и напоследок приведён обширный список инструментов для встраивания информации, которую нужно сокрыть и её чтения с пояснением недостатков некоторых из них. Вся эта статья рассчитана на получения новых знаний и на то, чтобы люди стали чаще использовать стеганографию для обеспечения сокрытия и проверки подлинности информации, так как её можно перехватить и видоизменить, чем злоумышленники часто пользуются. Ведь если использовать стеганографию в качестве подписи документов или фотографий можно проверить достоверность и подлинность этой информации, то есть была ли информация, действительно отправленная определённым человеком или же она была перехвачена.

Библиографический список

1. Грибунин В. Г., Костюков В. Е.; Мартынов А. П., Николаев Д. Б., Фомченко В. Н., Стеганографические системы. Критерии и методическое обеспечение: Учебно-методическое пособие / Под редакцией доктора технических наук В. Г. Грибунина, Саров: ФГУП "РФЯЦ-ВНИИЭФ", 2016.

2. Мухамедович Э. Г. и Ивановна Н. П., Лекции по теории информации. Учебно-методическое пособие., Москва: МФТИ: "Физматлит", 2007.

3. V. Korzhik, H. Imai, J. Shikata, G. Morales Luna and E. Gerling, "On the Use of Bhattacharya Distance as a Measure of the Detetability of Steganographic System," 2008, p. 23—32.

4. Конахович Г. Ф., Пузыренко А. Ю., Компьютерная стеганография. Теория и практика., К.: МК-Пресс, 2006.

5. Рябко Б. Я. и Фионов А. Н., Основы современной криптографии и стеганографии. — 2-е изд., М.: Горячая линия — Телеком, 2013.

6. Грибунин В. Г., Оков И. Н., Туринцев И. В., Цифровая стеганография, М.: Солон-Пресс, 2002.

УДК 004.056; ГРНТИ 81.93.29

ТЕХНИЧЕСКИЕ МЕРЫ ЗАЩИТЫ ПЕРСОНАЛЬНЫХ ДАННЫХ В РОССИЙСКОЙ ФЕДЕРАЦИИ

С.А. Правкин

Рязанский государственный радиотехнический университет имени В.Ф. Уткина, Российская Федерация, Рязань, spabout@yandex.ru

Аннотация. Статья содержит анализ технических мер защиты персональных данных в Российской Федерации: приведены нормативно-правовые акты, устанавливающие состав и содержание технических мер защиты персональных данных; сформулированы актуальные угрозы безопасности персональных данных, список потенциальных нарушителей, состав используемых мер технической защиты персональных данных, краткий перечень сертифицированных программно-аппаратных средств, способных самостоятельно или совместно обеспечить безопасность персональных данных.

Ключевые слова: технические меры защиты, персональные данные (ПДн), безопасность ПДн.

TECHNICAL MEASURES FOR THE PROTECTION OF PERSONAL DATA IN THE RUSSIAN FEDERATION

S.A. Pravkin

Ryazan State Radio Engineering University named after V.F. Utkin, Russia, Ryazan, spabout@yandex.ru

The summary. The article contains an analysis of technical measures to protect personal data in the Russian Federation: the regulatory acts that establish the composition and content of the technical measures to protect personal data, the current threats to the security of personal data, a list of potential violators, the composition of the used measures of technical protection of personal data, a brief list of certified hardware and software, capable of independent or jointly ensure the security of personal data.

Keywords: technical protection measures, personal data (PD), PD security.

Обеспечение безопасности персональных данных (далее – ПДн) одинаково значимо, как для гражданина, так и для государства (Российской Федерации). Высокая интеграция информационных технологий в процессы сбора, хранения, обработки и передачи ПДн, законодательство в сфере обеспечения безопасности ПДн, быстрое увеличение числа операторов ПДн и количества обрабатываемых ими ПДн, а главное регулярный рост возможного ущерба от массового несанкционированного использования ПДн сделали технические меры защиты – обязательным средством обеспечения безопасности ПДн в Российской Федерации (далее – РФ). Важность анализа технических мер защиты персональных данных в РФ в рамках данной работы можно обосновать также тем, что согласно «Доктрине информационной безопасности Российской Федерации», утвержденной Указом Президента РФ от 5 декабря 2016 г. № 646, «развитие в Российской Федерации отрасли информационных технологий .., а также совершенствование деятельности производственных, научных и научно-технических организаций по разработке, производству и эксплуатации средств обеспечения информационной безопасности, оказанию услуг в области обеспечения информационной безопасности» является национальным интересом РФ в сфере информационных технологий.

Нормативно-правовые акты, устанавливающие состав и содержание технических мер защиты ПДн

Основными нормативно-правовыми актами (далее – НПА), которые устанавливают состав и содержание технических мер защиты ПДн являются два федеральных закона, четыре постановления правительства, три приказа (два ФСТЭК России и один ФСБ России), они изображены на рисунке 1.

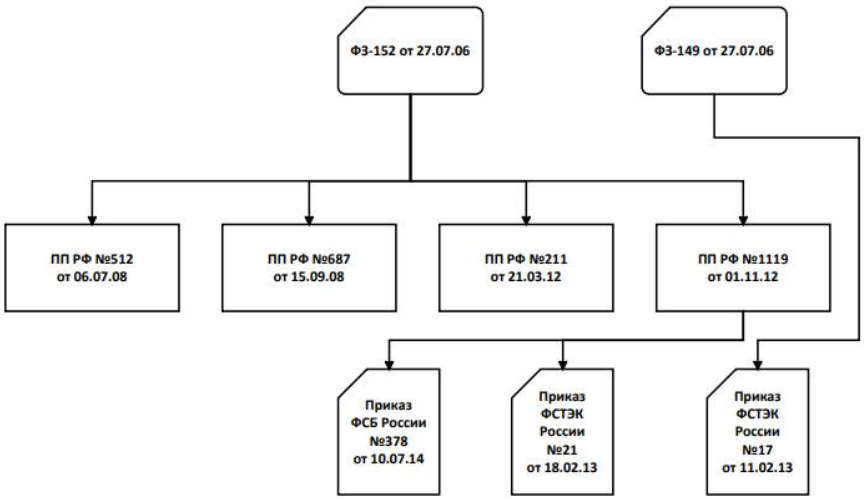


Рис. 1. Схема НПА, которые устанавливают технические меры защиты ПДн.

Актуальные угрозы безопасности ПДн

Термин «актуальные угрозы безопасности ПДн» в данной статье следует рассматривать, как частный результат анализа потенциальных нарушителей безопасности ПДн, отчетов о расследованиях утечек персональных данных и иных экспертных документов, а также НПА, которые можно отнести к сфере защиты ПДн в РФ. Систематизация сведений об актуальных угрозах и нарушителях безопасности ПДн в данном разделе проведена на основе методических рекомендаций, изложенных в "Методике оценки угроз безопасности информации" утвержденной ФСТЭК России 5 февраля 2021 года.

К совокупности условий и факторов, определяющих высокую потенциальную или реально существующую опасность нарушения безопасности ПДн относятся угрозы, представленные на рисунке 2.

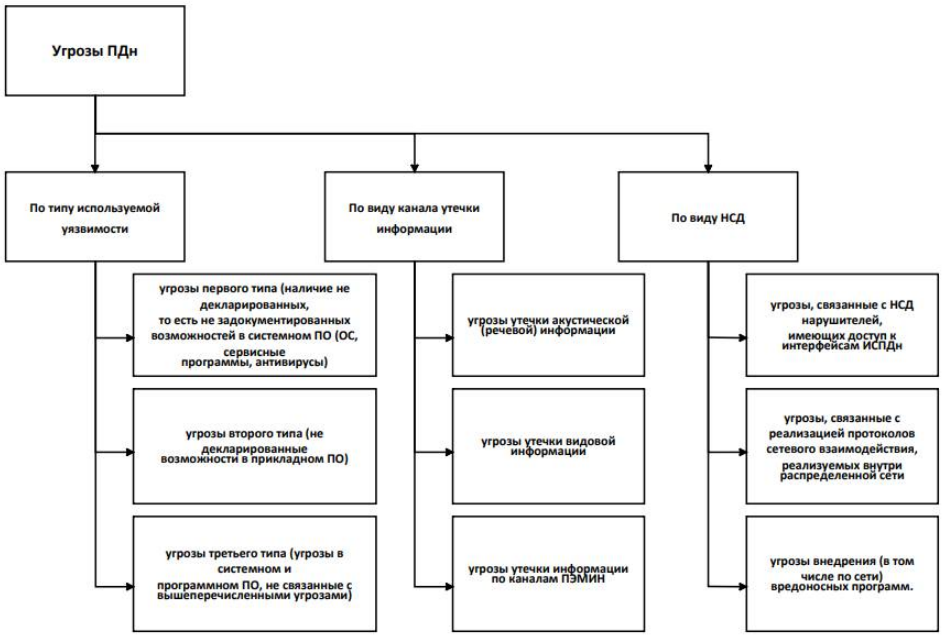


Рис. 2. Схема угроз ПДн.

Частными видами перечисленных угроз являются:

- НСД сотрудников в отношении ПДн, баз данных (несанкционированные запросы, копирование, предоставление НСД и изменение данных);
- сложные атаки хакерских преступных группировок (использоваие узкоспециализированных подходов к эксплуатации уязвимостей, программных и аппаратных закладок и т.п. вплоть до похищения сотрудников, имеющих доступ к ПДн, или же их родных для давления);
- сложные атаки разведывательных служб иностранных государств (повторение методов, применяемых преступными группировками, использование высококлассных специалистов в узких областях, создание аппаратных или программных закладок на самых ранних этапах изготовления устройст и программного обеспечения, соответственно);
- атаки на поставщиков услуг по обеспечению безопасности ПДн (получение доступа к пулу клиентской информации, включая архитектуру веб-сервисов, систем безопасности, версий используемого ПО, параметров реагирования на инцидент средвами управления информационной безопасностью и т.д.);
- «цифровой терроризм [1], [2]».

Потенциальные нарушители безопасности ПДн

Систематизация сведений об актуальных угрозах и нарушителях безопасности ПДн в данном разделе, как и в предыдущем, проведена на основе методических рекомендаций, изложенных в «Методике оценки угроз безопасности информации» утвержденной ФСТЭК России 5 февраля 2021 года.

Актуальными потенциальными нарушителями безопасности ПДн в тройке самых распространенных являются и в перспективе сохранят свои позиции:

- физическое лицо (хакер);
- преступные группы (два лица и более, действующие по единому плану);
- специальные службы иностранных государств.

Состав мер технической защиты ПДн

Мерами технической (программно-аппаратной) защиты ПДн согласно Приказу ФСТЭК России от 18 февраля 2013 г. №21 являются:

- идентификация и аутентификация субъектов доступа и объектов доступа;
- управление доступом субъектов доступа к объектам доступа;
- ограничение программной среды;
- защита машинных носителей информации, на которых хранятся и (или) обрабатываются персональные данные (далее - машинные носители персональных данных);
- регистрация событий безопасности;
- антивирусная защита;
- обнаружение (предотвращение) вторжений;
- контроль (анализ) защищенности персональных данных;
- обеспечение целостности информационной системы и персональных данных;
- обеспечение доступности персональных данных;
- защита среды виртуализации;
- защита технических средств;
- защита информационной системы, ее средств, систем связи и передачи данных;
- выявление инцидентов (одного события или группы событий), которые могут привести к сбоям или нарушению функционирования информационной системы и (или) к возникновению угроз безопасности персональных данных (далее - инциденты), и реагирование на них;

- управление конфигурацией информационной системы и системы защиты персональных данных».

Типовыми программно-аппаратными решениями, соответственно, реализующими перечисленные меры считаются:

1) встроенные механизмы безопасности ОС, прикладного ПО, контроллеров, активного сетевого оборудования, «навесная защита»;

2) встроенные механизмы безопасности ОС, прикладного ПО, контроллеров, активного сетевого оборудования, «навесная защита»;

3) встроенные механизмы безопасности ОС, прикладного ПО, контроллеров, активного сетевого оборудования, «навесная защита», эмулятор среды функционирования ПО (песочница);

4) встроенные механизмы безопасности ОС, прикладного ПО, контроллеров, активного сетевого оборудования, «навесная защита», средства контроля съемных носителей информации, средства инвентаризации машинных носителей информации, физическое ограничение доступа к портам ввода-вывода, DLP-системы;

5) средства мониторинга показателей функционирования компонентов информационных систем, управления активами и объектами защиты, анализа защищенности и обнаружения вторжений, системы управления информационной безопасностью и событиями безопасности, межсетевые экраны, DLP-системы;

6) средства антивирусной защиты электронных почтовых систем, антивирусы, антивирусные расширения для браузеров;

7) средства обнаружения, предотвращения вторжений, обнаружения атак, автоматизации процессов управления информационной безопасностью;

8) встроенные механизмами обеспечения информационной безопасности операционных систем, прикладного программного обеспечения, программируемых логических контроллеров, активного сетевого оборудования и т.п., сканеры анализа защищенности, средства контроля действий пользователей, в частности DLP – системы;

9) встроенные механизмами обеспечения информационной безопасности операционных си-стем, прикладного программного обеспечения, программируемых логических контроллеров, активного сетевого оборудования и т.п., сканеры анализа защищенности, средства контроля действий пользователей, в частности DLP – системы;

10) использование резервирования основных технических средств, резервирование каналов передачи информации, отказоустойчивые конфигурации, средства мониторинга показателей функционирования компонентов информационных систем, отправка событий на сервера регистрации по протоколу Syslog;

11) «средства резервного копирования виртуальных машин, антивирусной защита, мониторинга событий безопасности виртуальной инфраструктуры, разграничения доступа внутри виртуальной инфраструктуры, защита данных внутри виртуальных машин, сетевой защита внутри виртуальной машины, обнаружение вторжений внутри виртуальной инфраструктуры, контроля уязвимостей внутри виртуальной инфраструктуры, аудита действий привилегированных пользователей [3]»;

12) исключение (в данном случае применяются организационные и инженерно-технические меры защиты);

13) исключение (в данном случае применяются организационные и инженерно-технические меры защиты);

14) специализированные инструменты для разработки ролевой модели, позволяющие в автоматизированном режиме контролировать выполнение требований, использование встроенных средствам ПО оборудования и подсистемам идентификации и аутентификации системного и прикладного ПО, средства обнаружения, предотвращения вторжений, обнаружения атак, межсетевые экраны, эмулятор среды функционирования ПО (песочница).

15) средства автоматизации процессов управления информационной безопасности, резервного копирования и другие.

Сертифицированные программно-аппаратные средства защиты ПДн

Для реализации технических мер защиты ПДн до 80% можно использовать следующие наборы программно-аппаратных средства:

А) Программный комплекс средств защиты информации «Secret Net Studio» (версии 8.8);

Б) Программно-аппаратный комплекс «DATAPK»;

В) Система «InfoWatch ARMA»;

Г) Экосистема «UserGate SUMMA».

В целях защиты прав и свобод граждан, национальных интересов государства в Российской Федерации общественными институтами принимаются значительные усилия в области совершенствования всесторонней защиты ПДн, как следствие растет качество услуг и продуктов, предоставляемых в сфере реализации мер технической защиты ПДн. Но также согласно данным отчет об исследовании утечек информации ограниченного доступа в I половине 2022 года компании «InfoWatch» по количеству утечек ПДн РФ занимает одну из лидирующих позиций в мире.

Если провести параллели между выявленными актуальными угрозами, потенциальными нарушителями, нормами установленными НПА, сертифицированными программно-аппаратными средствами защиты ПДн и статистикой по количеству утечек, можно сделать вывод о том, что несмотря на наличие НПА, материально-технической базы в РФ частные и государственные операторы ПДн в недостаточной степени реализуют мероприятия по обеспечению защиты ПДн с использованием технических мер защиты.

Библиографический список

1. Степкин К.А., Федосов А.В. Актуальные вопросы противодействия цифровизации (информатизации) террористической активности [Электронный документ]. — URL: https://cyberleninka.ru/article/n/aktualnye-voprosy-protivodeystviya-tsifrovizatsii-informatizatsii-terrroristicheskoy-aktivnosti/viewer (дата обращения: 24.02.2023).

2. Кибертерроризм: виды атак, хакеров и их влияния на цели [Электронный документ]. — URL: https://www.securitylab.ru/analytics/534885.php (дата обращения: 24.02.2023).

3. Игнатов, С. Д. Обеспечение защиты информации в виртуализированной инфраструктуре / С. Д. Игнатов, А. А. Быстров //Молодой ученый. — 2021. — № 24 (366). — С. 30-34. — URL: https://moluch.ru/archive/366/82146/ (дата обращения: 25.02.2023).

УДК 004.056; ГРНТИ 81.93.29

ВЫБОР ОРГАНИЗАЦИОННЫХ И ТЕХНИЧЕСКИХ МЕР ЗАЩИТЫ ИНФОРМАЦИИ ДЛЯ АВТОМАТИЗИРОВАННОГО РАБОЧЕГО МЕСТА, ПОДКЛЮЧЕННОГО К СЕТИ «ИНТЕРНЕТ»

А.Г. Агафонов, Н.А. Колесенков

Аннотация. Рассмотрена нормативная правовая база Российской Федерации определяющая организационные и технические меры защиты информации в информационных системах. Определен алгоритм выбора мер защиты информация для автоматизированного рабочего места, подключенного к сети «Интернет» и обрабатывающего информацию соответствующей значимости.

Ключевые слова: информация, защита информации, информационная система, автоматизированное рабочее место, организационные меры защиты информации, технические меры защиты.

SELECTION OF ORGANIZATIONAL AND TECHNICAL INFORMATION PROTECTION MEASURES FOR AN AUTOMATED WORKPLACE CONNECTED TO THE INTERNET

A.G. Agafonov, N.A. Kolesenkov

Abstract. The normative legal base of the Russian Federation defining organizational and technical measures of information protection in information systems is considered. The algorithm for selecting information protection measures for an automated workplace connected to the Internet and processing information of appropriate significance is determined.

Keywords: information, information protection, information system, automated workplace, organizational information protection measures, technical protection measures.

Организационно-технические меры защиты информации – это одни из главных элементов защищенной информационной системы. Когда их выбор соответствует актуальным угрозам безопасности информации, то они являются гарантом надёжного и безопасного функционирования системы, в которой установлены.

В рассматриваемом вопросе главное не только защитить объект защиты, но и не использовать в подсистеме защиты не нужные (избыточные) механизмы защиты. Как неполная защита, так и избыточная, влекут за собой серьезный вред, обычно финансовый или имиджевый, поскольку средства защиты информации обладают достаточно высокой стоимостью, а отсутствие требуемых механизмов в подсистеме защиты влечет за собой потери в результате действий нарушителей. Вместе с этим, если в результате действий нарушителя был ущерб собственнику информации, в результате ее утечки, например персональных данных, которые обрабатываются в данной системе, тогда на оператора этой системы, в соответствие с действующим законодательством будут наложены определённые санкции, как правило штрафные.

Наиболее важными и значимыми для общества и государства являются государственные информационные системы. Следовательно, рассматривать выбор организационно-технических мер будем в контексте именно таких информационных систем. От их безопасного функционирования непосредственно зависит качество предоставляемых государственных услуг.

Организационно-технические меры защиты информации служат для исключения таких угроз безопасности, как нарушения одного или нескольких основных свойств безопасности информации, а именно:

1. Рассматриваемые меры предотвращают неправомерный (несанкционированный) доступ к информации на этапах её обработки, в том числе при проведении ремонтных и других работ, копирование, предоставление или распространение (конфиденциальность);

2. Рассматриваемые меры направлены на недопущение неправомерного уничтожения или модифицирования информации (целостность);

3. Рассматриваемые меры препятствуют блокированию информации (доступность).

Главные понятия в этой работе – это «Организационная защита информации» и «Техническая защита информации». Определим их:

Организационная защита информации – это меры, которые направлены на регламентацию функционирования информационной системы, работы сотрудников, взаимодействия с системой пользователей.

Базовые организационные меры:

– формирование политики безопасности;

– регламентация доступа в помещения;

– регламентация допуска сотрудников к использованию ресурсов информационной системы;

– определение ответственности в случае несоблюдения требований информационной безопасности.

Техническая защита информации – это некриптографические меры по применению технических, программных и программно-технических средств защиты информации, которая подлежит защите в соответствии с действующим законодательством. Важно обратить внимание, что сейчас, более всего техническая защита направлена на предотвращение неправомерного (несанкционированного) доступа к защищаемой информации.

Несанкционированный доступ – это доступ с нарушением установленных правил доступа к информации и (или) ресурсам автоматизированной информационной системы [6].

Организационно-технические меры защиты информации, реализуемые в рамках подсистемы защиты, в зависимости от угроз безопасности информации, структурно-функциональных характеристик и используемых информационных технологий должны обеспечивать:

1. Аутентификацию и идентификацию объектов/субъектов доступа;
2. Управлять доступом субъектов к объектам доступа;
3. Ограничивать программную среду;
4. Защищать носители (машинные) информации;
5. Регистрировать событий безопасности;
6. Антивирусную защиту;
7. Обнаруживать и предотвращать вторжения;
8. Контролировать и анализировать защищенность информации;
9. Обеспечивать целостность и доступность информации;
10. Защищать технических средств;
11. Защищать средства информационной системы [4].

Информационная система – это совокупность содержащейся в базах данных информации и обеспечивающих ее обработку информационных технологий и технических средств. Таким образом под государственной информационной системой будем понимать федеральные информационные системы и региональные информационные системы, созданные на основании соответственно федеральных законов, законов субъектов Российской Федерации, на основании правовых актов государственных органов [1].

Определение понятия «Автоматизированная система» даётся в ГОСТ Р 59853-2021. Автоматизированная система – это система, состоящая из комплекса средств автоматизации, реализующего информационную технологию выполнения установленных функций, и персонала, обеспечивающего его функционирование (АС)[2].

В этом же документе даётся определение понятия «Автоматизированное рабочее место». Автоматизированное рабочее место – Программно-технический комплекс автоматизированной системы, предназначенный для автоматизации деятельности определенной категории пользователей или определенного вида деятельности [2]. Как правило, под автоматизированным рабочим местом понимается стационарный компьютер или ноутбук, устанавливаемый в границах помещений и (или) сооружений организаций или предприятий в приделах их контролируемой зоны, выход за которую возможен если автоматизированное рабочее место имеет подключение к сети «Интернет».

Подключение автоматизированного рабочего места к сети «Интернет» расширяет перечень угроз безопасности информации в части несанкционированного доступа, например: выявление паролей, «Отказ в обслуживании», удалённый запуск приложений, внедрение по сети вредоносных программ [3].

Исходя из рассмотренных выше определений, можно сделать вывод, что государственная информационная система является автоматизированной информационной системой, взаимодействие в которой осуществляется автоматизированными рабочими местами, операторами которых является персонал этой системы.

Так же согласно названному документу [4] выбор защитных мер для их реализации в рамках системы защиты информации включает:

– формирование набора защитных мер (базовых) для определённого класса защищенности. Это производится в соответствии с базовыми наборами мер защиты информации, приведенными в приложении номер 2 к указанному документу;

– уточнение модели угроз и нарушителей;

– уточнение базового набора защитных мер применительно к реальной системе;

Как следует из вышесказанного ключевую роль в выборе тех или иных мер защиты играет уровень значимости информации, циркулирующей в этой системе и соответствующий этому уровню класс защищённости, при учёте масштаба информационной системы (федеральный, региональный, объектовый). Определение данных параметров выполняется в соответствии с приведёнными данными в приложении номер 1 документа [4].

Следует отметить, что уровень значимости информации определяется согласно степени возможного ущерба (высокий ущерб, средний и низкий) для обладателя информации и (или) оператора.

Когда был определён класс защищённости нужно приступить к определению базовых мер защиты, которыет в последующем должен быть обязательно дополнен моделью угроз безопасности информации и моделью нарушителя.

Для этого используется приложение номер 2, где для каждого из пунктов того, что должны обеспечивать организационные и технические меры приведены базовые меры защиты информации, например:

– для каждого из трёх классов защищённости в разделе «Идентификация и аутентификация субъектов доступа и объектов доступа (ИАФ)» определена такая мера, как «I. Идентификация и аутентификация пользователей, являющихся работниками оператора» (ИАФ.1), а далее для второго и первого класса защищённости определяется мера «Идентификация и аутентификация устройств, в том числе стационарных, мобильных и портативных» (ИАФ.2);

– для первого класса защищённости в разделе «III. Ограничение программной среды (ОПС)», определена мера «Управление запуском (обращениями) компонентов программного обеспечения, в том числе определение запускаемых компонентов, настройка параметров запуска компонентов, контроль за запуском компонентов программного обеспечения» и т. д. [4]

Таким образом, последовательно анализируя все разделы рассматриваемого приложения, формируется список базовых мер защиты информации, где под каждый из пунктов в дальнейшем будет принято конкретное средство защиты информации.

Одним из дополнений к информационной системе в случае, когда она имеет доступ к информационно-телекоммуникационной сети «Интернет», является то, что в таких случаях, требуется использовать маршрутизаторы, сертифицированные на соответствие требованиям по безопасности информации (в части реализованных в них функций безопасности) [4].

Согласно ГОСТ Р 53632-2009 «Показатели качества услуг доступа в интернет. Общие требования» маршрутизатор – это устройство, которое пересылает пакеты между сетями.

Также к организационно-техническим мерам зашиты можно отнести использование многофакторной аутентификации – это метод предоставления доступом к рабочему месту, оператору которого предоставляется доступ в систему только после того, как тот успешно предъявит несколько отдельных доказательств механизму аутентификации. Например, пароль и магнитный ключ. Пароль человек может запомнить, а магнитный ключ будет выдаваться уполномоченным сотрудником в установленном внутреннем регламенте порядке. Это поможет избежать ситуаций, когда вредоносное программное обеспечение (вирус, троян и т.п.), распространяемое через сеть «Интернет», будет пытаться получить доступ на целевые ресурсы. Таким образом, даже если вредоносная программа сможет действовать в операци-

онной системе от имени администратора ей будет необходимо, что бы оператор задействовал свой физический ключ.

В заключение можно сказать, что выбор как организационных, так и технических мер защиты информации является неотъемлемым в процессе разработки информационной системы, особенно в тех случаях, когда в ней циркулирует информация конфиденциального характера и персональные данные.

Верный выбор является залогом безопасной работы системы, что исключает как финансовый, так и имиджевый ущерб. Если автоматизированное рабочее место подключено к сети «Интернет», то необходимо предпринимать дополнительные меры по устранению угроз, добавляемых этим фактором, особенно угроз несанкционированного доступа.

Большинство угроз в автоматизированных информационных системах (АРМ) устраняет СЗИ «Secret Net Studio», являясь универсальным многофункциональным средством информационной защиты, которое обеспечивает многоуровневую защиту и соответствует всем требованиям, предъявляемым к средствам защиты информации, имеет соответствующую сертификацию ФСТЭК России.

Библиографический список

1. Федеральный закон от 27.07.2006 №149-ФЗ «Об информации, информационных технологиях и о защите информации» [Электронный ресурс]. – Режим доступа: http://www.consultant.ru/document/cons_doc_LAW_61798

2. ГОСТ Р 59853-2021 «Информационные технологии. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения».

3. Методический документ ФСТЭК России «Базовая модель угроз безопасности персональных данных при их обработке в информационных системах персональных данных» [Электронный ресурс]. – Режим доступа: https://fstec.ru/tekhnicheskaya-zashchita-informatsii/dokumenty/114spetsialnye-normativnye-dokumenty/379bazovaya-model-ugroz-bezopasnosti-perso-nalnykh-dannykh-pri-ikh-obrabotke-v-informatsionnykh-sistemakh-personalnykh-dannykh-vypiska-fstek-rossii2008god?ysclid=leesi06c18945749599

4. Приказ ФСТЭК России от 11.02.2013 № 17 «Об утверждении требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах».

5. Шаньгин В.Ф. Защита компьютерной информации. Эффективные методы и средства / Шаньгин В. Ф. – М.: ДМК Пресс, 2008. – 544 с.:ил.

6. ГОСТ Р 53114-2008 «Защита информации. Обеспечение информационной безопасности в организации. Основные термины и определения».

СОДЕРЖАНИЕ

СОВРЕМЕННЫЕ ТЕХНОЛОГИИ В НАУКЕ И ОБРАЗОВАНИИ

Научное издание

В 10 томах

Том 3

Под общей редакцией О.В. Миловзорова.

Подписано в печать 15.06.23. Формат 60x84 1/8.
Бумага офсетная. Печать офсетная.
Гарнитура «Times New Roman».
Усл. печ. л..
Тираж 100 экз. Заказ №.

ISBN 978-5-7722-0376-7

Рязанский государственный радиотехнический университет,
Редакционно-издательский центр РГРТУ,
390005, г. Рязань, ул. Гагарина, д. 59/1.