

Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»



**Информационные технологии и системы 2021
(ИТС 2021)**

Материалы международной научной конференции
(БГУИР, Минск, Беларусь, 24 ноября 2021)

**Information Technologies and Systems 2021
(ITS 2021)**

Proceeding of The International Conference
(BSUIR, Minsk, Belarus, 24th November 2021)

Минск БГУИР 2021

УДК 004
ББК 32.81
И74

Редакционная коллегия:

Л. Ю. Шилин (главный редактор), А. А. Иванюк, С. В. Колосов,
В. С. Муха, В. В. Голенков, А. В. Марков, Д. П. Кукин, А. А. Навроцкий,
В. И. Журавлев, А. Ф. Трофимович, Н. В. Гракова, Д. В. Шункевич,
А. Б. Гуринович (ответственный секретарь)

Информационные технологии и системы 2021 (ИТС 2021) =
И74 Information Tehnologies and Systems 2021 (ITS 2021) : материалы
международной научной конференции, Минск, Беларусь, 24 ноября 2021 г.
/ Л. Ю. Шилин [и др.]. – Минск : БГУИР, 2021. – 248 с.
ISBN 978-985-543-013-2.

Сборник включает прошедшие рецензирование доклады международной научной конференции «Информационные технологии и системы 2021» (ИТС 2021).

Сборник предназначен для преподавателей высших учебных заведений, научных сотрудников, студентов, аспирантов, магистрантов, а также для специалистов предприятий в сфере IT-технологий.

Материалы сборника одобрены организационным комитетом и печатаются в авторской редакции.

УДК 004
ББК 32.973.202

ISBN 978-985-543-013-2

Белорусский государственный университет
информатики и радиоэлектроники», 2021

ПРЕДИСЛОВИЕ

24 ноября 2021 года в городе Минске прошла международная научная конференция «Информационные технологии и системы» (ИТС 2021), организованная факультетом информационных технологий и управления учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», базовой организации государств-участников СНГ по высшему образованию в области информатики и радиоэлектроники.

В течение дня на пленарном и пяти секционных заседаниях было представлено 102 доклада по следующим направлениям:

Секция «Системы управления»:

- моделирование систем управления;
- автоматизированные исследования систем управления;
- проектирование систем управления;
- микропроцессорные системы управления;
- информационные технологии в системах управления;
- системы передачи и защиты дискретной информации;
- системы телемеханики;
- элементы и устройства систем управления.

Секция «Интеллектуальные информационные технологии»:

- проектирование баз знаний;
- интеллектуальный поиск информации;
- интеллектуальные решатели задач;
- пользовательские интерфейсы интеллектуальных систем.

Секция «Проектирование встраиваемых систем»:

- проектирование и быстрое прототипирование встраиваемых систем на ПЛИС;
- методологии проектирования IP-компонент встраиваемых систем;
- программная и аппаратная защита IP-компонент;
- контролепригодное проектирование и тестирование встраиваемых систем;
- реконфигурируемые встраиваемые системы.

Секция «Автоматизированные системы обработки информации»:

- математические модели и методы обработки информации: статистические, численные, логические, графовые, нейросетевые и другие;
- анализ многомерных данных;
- методологические проблемы и технологии подготовки специалистов в области систем обработки информации;
- технологии и стандарты программирования и проектирования систем обработки информации;
- системы и средства совместной работы пользователей;
- защита информации в автоматизированных системах обработки информации.

Секция «Математическое моделирование и компьютерная графика»:

- методы решения уравнений математической физики;
- методы нелинейной оптимизации;
- модели и методы тепло и массопереноса;
- математические модели и методы компьютерной графики;
- аппаратное обеспечение компьютерной графики;
- алгоритмы и методы интерактивных программных продуктов.

В работе конференции приняли участие учёные и специалисты Республики Беларусь, Российской Федерации, Украины.

Докладчики представляли как высшие учебные заведения, в их числе: БГУ, БГУИР, БГТУ, ПГУ, БНТУ, Белорусская государственная академия связи, Военная академия Республики Беларусь, Национальный университет пищевых технологий, Белорусско-Российский университет, Финансовый университет при Правительстве Российской Федерации, Уфимский государственный нефтяной технический университет – так и институты и коммерческие организации: Объединенный институт проблем информатики НАН РБ, Институт информационных технологий, Республиканский научно-практический центр «Кардиология», Республиканский центр медицинской реабилитации и бальнеолечения, Городской клинический родильный дом № 2, Институт проблем управления им. В.А. Трапезникова РАН, ООО «СК хайникс мемори солюшнс Восточная Европа», ОАО «Савушкин продукт».

Оргкомитет благодарит всех, кто счел возможным поделиться результатами своей деятельности и принял участие в работе конференции.

ORGANISING COMMITTEE

Prof.Dr.	Vadim Bogush	Rector of The Belarusian State University of Informatics and Radioelectronics
PhD	Maksim Davydov	First Vice-Rector of BSUIR
PhD	Evgeni Shneiderov	Vice-Rector for Education of BSUIR
PhD	Viktor Stempitsky	Vice-Rector for Research of BSUIR
	Dmitry Kuznetsov	Vice-rector for Personal Development of BSUIR
PhD	Victor Rybak	Vice-Rector for Education of BSUIR
Prof.Dr.	Leonid Shilin	Dean of The Faculty of Information Technologies and Control of BSUIR

PROGRAMME COMMITTEE

Prof.Dr.	Vladimir Mukha	Professor of Information Technologies in Automated Systems Department of BSUIR
Prof.Dr.	Vladimir Golenkov	Professor of Intelligent Information Technologies Department of BSUIR
Prof.Dr.	Vladimir Golovko	Head of Intelligent Information Technologies Department of BrSTU
Prof.Dr.	Alexander Ivaniuk	Professor of Computer Science Department of BSUIR
Prof.Dr.	Stanislav Kolosov	Professor of Computational Methods and Programming Department of BSUIR
PhD	Dmitry Kukin	Head of Computational Methods and Programming Department of BSUIR
PhD	Alexander Markov	Head of Control Systems Department of BSUIR
PhD	Anatoly Navrotsky	Head of Information Technologies in Automated Systems Department of BSUIR
PhD	Daniil Shunkevich	Head of Intelligent Information Technologies Department of BSUIR
PhD	Vadim Zhuravliov	Head of Fundamental Electrical Engineering Department of BSUIR

Пленарные доклады

Plenary session

ПРОЕКТИРОВАНИЕ И ВЕРИФИКАЦИЯ КОНТРОЛЛЕРОВ SSD-ДИСКОВ

Авдеев Н. А., Глебов С. К., Заливако С. С., Иванюк А. А., Пинчуков Д. А., Семенов В. В.,
Цибулько Ю. В.

ООО «СК хайникс мемори солишнс Восточная Европа»

Минск, Республика Беларусь

E-mail: {nikolai.avdeev, sergey.glebov, sergey.zalivako, alexander.ivaniuk, dmitry.pinchukov, vladimir.semenov,
yury.tsybulka}@sk.com

Потребность в системах хранения данных в настоящее время непрерывно растёт. Направление SSD-дисков в системах хранения данных в последнее десятилетие развивается наиболее интенсивно. С ростом объема хранимых данных на одном диске и скорости передачи данных растет и сложность контроллеров SSD-дисков, реализующих протокол передачи данных с компьютером и обеспечивающих запись и считывание данных с NAND флеш-памяти. В данной работе рассматриваются основные особенности проектирования и верификации современных контроллеров SSD-дисков, на основе опыта разработки таких устройств резидентом ПВТ ООО «СК хайникс мемори солишнс Восточная Европа» (SKHMS E).

ВВЕДЕНИЕ

Современный мир невозможно представить без использования информационных технологий и глобальной сети Интернет. Как известно, объём хранимых данных в Интернете экспоненциально растёт и поэтому спрос на системы хранения данных в мире сохранится в будущем. В последнее десятилетие технология твёрдотельных накопителей (Solid State Drive, SSD – далее SSD-диск) активно совершенствуется. SSD-диски имеют ряд преимуществ перед HDD-дисками: более быстрый отклик на действия пользователя, беззвучная работа, компактные размеры, меньшее энергопотребление, отсутствие механических подвижных частей, что способствует большей удароустойчивости. Хранение данных в SSD-диске происходит в 3D NAND флеш-памяти [1], представляющей собой электрически стираемое перезаписываемое постоянное запоминающее устройство. Запоминающая ячейка флеш-памяти основана на МОП-транзисторе с плавающим затвором. Накопитель устроен таким образом, что ячейки памяти располагаются не только в одной плоскости, но и вертикально друг над другом, составляя несколько десятков уровней [2]. В современных накопителях одна такая ячейка способна сохранять до четырех бит данных [1]. Скорость записи данных достигает 3,2 ГБайт/с, скорость чтения – 3,5 ГБайт/с [3].

Компания SK hynix является одним из немногих производителей флеш-накопителей в мире, который может обеспечить полный цикл разработки и производства всех необходимых компонент внутри компании, а именно – разработку и производство 3D NAND флеш-памяти, DRAM, контроллеров и программного обеспечения накопителей, вспомогательного программного обеспечения.

История компании SK берет свое начало в 1983 году. Компания, которая в наши дни известна как третий по продажам производитель полупроводниковых приборов в мире, начинала с разработки и производства динамической памяти. На данный момент компания также разрабатывает и производит NAND флеш-память, КМОП-сенсоры, SSD и мобильные накопители на базе флеш-памяти и имеет множество исследовательских центров и производственных мощностей по всему миру.

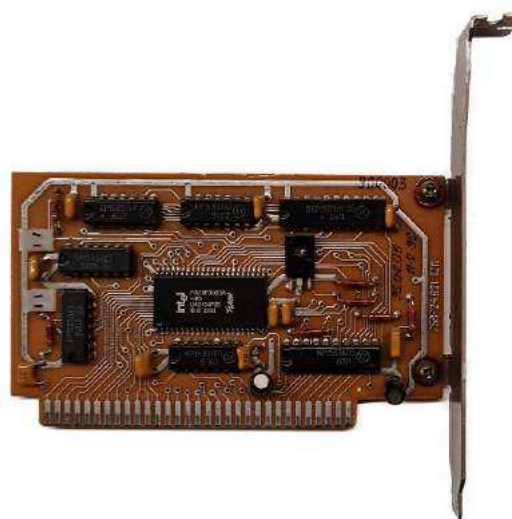


Рис. 1 – SSD-накопитель

Для обеспечения полного цикла разработки накопителей компания создала четыре исследовательских центра, которые расположены в Южной Корее, Соединенных Штатах Америки, Тайване и Беларуси. Отдельно стоит отметить, что появление исследовательского центра в Беларуси не случайно. Еще до прихода SK hynix на белорусский рынок, в стране уже существовала команда специалистов, имеющих опыт в создании

программного обеспечения для накопителей на базе NAND флеш-памяти, а первый накопитель на базе флеш-памяти был разработан и выпущен на территории Беларуси еще в 1996 году (см. рис. 1).

Процесс разработки флеш-накопителя может занять не один год и состоит из множества этапов работ, выполняемых командами четырех исследовательских центров. Белорусский центр принимает участие в таких этапах разработки как, проектирование аппаратуры контроллеров накопителей, формальная и функциональная верификация контроллеров, валидация контроллеров, разработка программного обеспечения накопителей, верификация программного обеспечения накопителей, разработка алгоритмов для последующего их использования в программном или аппаратном обеспечении флеш-накопителей. Кроме этого, белорусский офис отвечает за разработку сервисного программного обеспечения накопителей, ведет исследовательские работы и занимается подготовкой кадров при университетах Беларуси. В частности, белорусское подразделение SK hynix явилось инициатором организации совместной учебной лаборатории с БГУИР.

Как уже упоминалось, процесс разработки флеш-накопителя имеет большую продолжительность в силу его сложности. Контроллер SSD-диска представляет собой систему на кристалле (СнК), содержащую более 10 процессорных ядер и десятки аппаратных ускорителей, а интерфейсы работают на частотах в десятки гигагерц. Как следствие, все перечисленные этапы разработки имеют критическое влияние как на время выхода конечного продукта на рынок, так и на минимизацию рисков ошибок в программном и аппаратном обеспечении. Рассмотрим эти этапы подробнее.

I. УСТРОЙСТВО SSD-КОНТРОЛЛЕРА

Контроллер SSD-диска (SSD-контроллер) отвечает за управление данными, хранящимися в NAND флеш-памяти и обеспечивает связь с другими устройствами (хостом), будь то компьютер, цифровая камера или мобильный телефон. SSD-контроллер имеет сложную архитектуру СнК, в состав которой входят (см. рис. 2):

- хост-контроллер, включающий интерфейс протокола передачи данных (USB, Ethernet, PCIe, Unipro, CXL);
- процессорная подсистема, обычно состоящая из нескольких вычислительных ядер общего назначения (ARM, CXC, RISC-V);
- подсистема NAND флеш-контроллера;
- контроллеры внешней памяти различных типов (DDR, SDRAM);
- энергонезависимая память и/или средства долговременного хранения конфигурации СнК (EEPROM или флеш-память);

- широкий набор контроллеров вспомогательных периферийных интерфейсов (UART, SPI, I²C, GPIO, и т. п.);
- средства управления доменами частот и частотами, средства управления доменами питания, интеллектуальная логика сброса устройств и доменов;
- средства отладки и анализа производительности (DFT, JTAG, а также собственные интерфейсы);
- комплексная многоуровневая система коммуникации элементов в составе СнК (AXI, AHB, APB интерфейсы).

Хост-контроллер обеспечивает связь с внешним устройством по стандартному протоколу. В настоящее время наиболее используемым протоколом для SSD-дисков является NVMe-протокол [4] через PCIe интерфейс, а для мобильных устройств – UFS на базе UniPro [5].

Контроллер флеш-памяти обеспечивает запись и чтение данных в NAND флеш-накопитель, контроль количества циклов чтения/записи, обработку ошибок и прочие операции с данными. Чаще всего для ускорения операций чтения и записи флеш-контроллер является многоканальным и обеспечивает параллельную работу сразу с несколькими флеш-накопителями.

Для хранения данных в контроллере используется DDR-память, где сохраняются данные, полученные от хоста перед записью их во флеш-память, а также данные, вычитанные из флеш-памяти перед отправкой их на хост. Для хранения встроенного программного обеспечения (Firmware, FW) служит SRAM-память и загрузочная энергонезависимая память.

II. ФУНКЦИОНАЛЬНАЯ ВЕРИФИКАЦИЯ SSD-КОНТРОЛЛЕРА

Задача функциональной верификации заключается в том, чтобы доказать, что модель СнК или сложно-функционального блока (СФ-блока) позволяет успешно выполнять на ней требуемые задачи во всех допустимых условиях и режимах, предусмотренных спецификацией. Функциональная верификация SSD-контроллера разделяется на несколько уровней:

- уровень СФ-блоков (периферийные блоки, коммутаторы и сети, вычислительные процессоры, акселераторы и пр.);
- уровень подсистем (таких как хост и флеш-контроллеры, процессорная подсистема);
- уровень всей СнК.

Верификация на уровне СФ-блоков проводится с использованием классических UVM тестирующих программ [6]. На этом уровне максимально проверяется функционирование блока в соответствии с его спецификацией, получая 100% покрытие кода и полное функциональное покрытие. В большинстве случаев при этом время мо-

делирования всех тестов для СФ-блока не превышает нескольких часов.

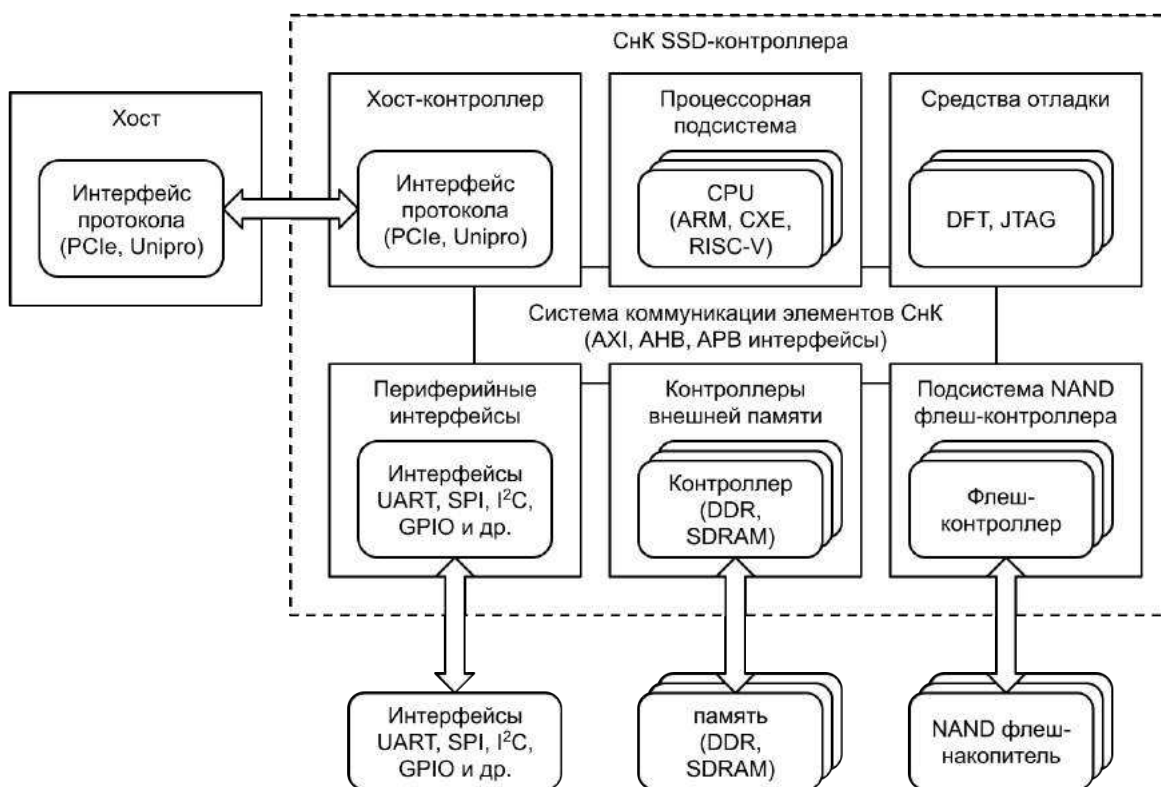


Рис. 2 – Типовая блок-схема СнК SSD-контроллера

При верификации на уровне подсистемы проверяется взаимодействие СФ-блоков между собой, а также работа основных функций, выполняемых подсистемой. Например, на уровне хост-контроллера следует проверить поддерживаемые команды протокола передачи данных, причем акцент делать на командах, которые имеют аппаратную реализацию в подсистеме, а не программную. Также нужно проконтролировать все сценарии взаимодействия встраиваемого ПО (со стороны процессора и других связанных подсистем) и хост-контроллера, проверить доступ к регистрам всех внутренних блоков, а также установку и сбрасывание всех прерываний, обработать сценарии сбросов подсистемы. На данном уровне время моделирования одного теста может занимать несколько часов, что значительно увеличивает требуемые вычислительные ресурсы для проведения ежедневных и еженедельных запусков регрессионного тестирования.

На уровне СнК верифицируется в первую очередь правильность взаимодействия всех блоков и подсистем, проверяется доступ ко всем регистрам в системе, проверяется базовый функционал, а также оценивается быстродействие системы при выполнении основных функций СнК. Особое внимание уделяется динамическим и формальным проверкам переходов между асин-

хронными доменами, подсистемам управления питанием и генераторам частот. Длительность моделирования одного теста на данном уровне варьируется от нескольких часов до нескольких дней, поэтому как только начинают проходить базовые тесты на данном уровне, то сразу приступают к следующему этапу верификации – валидации с помощью FPGA (например, на основе платформы прототипирования HAPS от фирмы Synopsys [18], а также ко-симуляции с помощью программно-аппаратных ускорителей моделирования, таких как Veloce [7], Palladium [8], Zebu [9].

III. СТАНДАРТИЗАЦИЯ ПРОЦЕССА ВЕРИФИКАЦИИ

Можно выделить несколько основных аспектов, которые нужно учитывать, чтобы процесс функциональной верификации был максимально эффективным.

Высокая скорость создания тестирующих программ (testbench), что достигается использованием стандартной структуры директорий проекта, переиспользование общих исходных кодов (базовые классы, пакеты, агенты, makefile и скрипты для запуска компиляции и симуляции проекта, и прочее), разработка собственных Verification intellectual property (VIP), а так-

же использование генераторов тестового окружения. Немаловажным является обмен знаниями и постоянное обучение инженеров.

Автоматизация создания и тестирования регистровых моделей и моделей памяти обеспечивается использованием специализированных генераторов, таких как Semifore CSRCompiler [10], позволяющих по одному исходному описанию регистров системы получать описание регистров для RTL в виде Verilog-кода, для HDL-моделирования в виде UVM регистровой модели, в виде заголовочных файлов для разработки встроенного ПО, а также документацию в различных форматах.

Верификация каждого отдельного СФ-блока, входящего в СнК, позволяет достичь 100% покрытия кода, сократить время моделирования и проверить максимальное число тестовых сценариев, что не отменяет проведение верификации на других уровнях: подсистем и СнК.

Параметризация тестирующей программы (Testbench) для возможности переиспользования в мульти-блочном режиме (когда тестируются несколько однотипных блоков одновременно). Также, при разработке тестирующей программы, необходимо учитывать возможность переиспользования эталонной модели СФ-блока на более высоком уровне (подсистемы или СнК), для этого она должна разрабатываться с учетом возможности работы в пассивном режиме, т. е. следить только за внешними портами блока и не «подсматривать» за внутренними сигналами блока – верификация «чёрного ящика».

Максимально возможное документирование кода и поддержание документации в актуальном состоянии. Для выполнения этой задачи необходимо использование общих шаблонов для документации, это позволяет стандартизировать и упростить процесс ее разработки. Также следует использовать средства для совместной разработки документации, например, Confluence [11], Mediawiki [12]. Использование программ документирования кода, таких как Doxygen [13], Natural Docs [14], WaveDrom [15], позволяет автоматизировать обновление документации.

Разработка верификационного плана обязательна. Необходимо унифицировать требования к описанию тестовых сценариев, их созданию и отслеживанию, определить обязательный набор метрик, организовать автоматический сбор всех видов покрытия и статуса выполнения тестов, а также сделать вывод статистики для последующего анализа состояния проекта. Можно выделить следующие основные задачи планирования верификации:

- определить набор свойств (функций) и тестов для верификации на каждом уровне (СФ-блоков, подсистемы, СнК);

- расставить приоритеты покрытия свойств, т. к. выделенного на верификацию времени может быть недостаточно для покрытия всех свойств;
- определить методы и средства верификации для разных групп свойств, при этом следует исключить дублирования верификации одних и тех свойств разными тестами, определиться какие свойства какими средствами могут быть проверены (динамические, статические тесты, FPGA) и на каких уровнях верификации какие из свойств проверять (уровень СФ-блоков, подсистемы, СнК);
- очень важно организовать взаимодействие между инженерами и командами, работающими на разных этапах и уровнях верификации;
- определить критерии успешного окончания верификации.

Критериями завершения верификации могут являться: выполнение всех пунктов верификационного плана, прохождение всех тестов без ошибок, достижение заданного уровня покрытия кода и функционального покрытия, успешное прохождение случайных тестов во время регрессионного тестирования длительное время.

Оценка функционального и кодового покрытия позволяют:

- оценить полноту верификации проекта;
- определить, какие существующие тестовые сценарии можно расширить для повышения покрытия;
- определить, какие дополнительные тесты необходимо разработать для повышения покрытия;
- своевременно определять «избыточные» тесты (которые не увеличивают покрытие) и исключать их;
- удостовериться в том, что тест проверяет необходимое свойство;
- оценить эффективность случайного теста.

Выполнение регулярного регрессионного тестирования позволяет своевременно выявить возникновение ошибки в базе проекта, а запрет на внесение изменений в базу проекта без прохождения специальной, обычно достаточно быстрой, регрессии защищает базу исходных кодов проекта от внесения ошибок. Для осуществления автоматического запуска регрессий могут использоваться программы Jenkins [16], BuildBot, Enterprise Manager (Cadence).

Эффективное использование вычислительных ресурсов, таких как серверы (ядра), оперативная память и лицензии на САПР и VIP, достигается использованием программ распределения задач в сети удалённых серверов, таких как RTDA, LSF и др.

Использование Verification intellectual property (VIP) позволяет сократить сроки ве-

рификации и уменьшить вероятность ошибок в проекте. При верификации SSD-контроллеров в основном используются VIP для протоколов PCIe, NVMe, UniPro, UFS и др. VIP могут включать как набор UVM-агентов, упрощающих разработку тестов, так и сами наборы тестов для выполнения полной проверки верифицируемого блока. Использование покупных VIP связано с рядом особенностей: число доступных лицензий обычно ограничено высокой стоимостью VIP, что может существенно влиять на длительность регрессии; при разработке SSD-контроллеров с поддержкой новых ревизий стандартов протоколов высока вероятность, что сроки обновления VIP не будут согласованы со сроками выполнения проекта и соответствующий VIP может содержать ошибки в реализации новых функций протокола; сроки исправления ошибок в VIP могут варьироваться от нескольких дней до месяцев. В связи с вышесказанным в долгосрочной перспективе целесообразно разрабатывать собственные VIP для верификационных компонентов, особенно для широко используемых в SnK протоколов, таких как UART, SPI, AXI, ANV, APB и прочих. Собственные VIP имеют открытый код, а значит обеспечивают более быстрое выявление и исправление ошибок, по сравнению с покупным VIP. В общем случае лучше выбирать различных разработчиков VIP и соответствующего IP (intellectual property) в SnK, например, если в проекте используется IP PCIe интерфейса от фирмы Synopsys, то лучше отдать предпочтение VIP от другой фирмы.

IV. РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ НАКОПИТЕЛЯ

Одним из основных компонентов флеш-накопителя, управляющим его работой, является программное обеспечение, известное как FTL (Flash Translation Layer).

FTL флеш-накопителя обеспечивает надежное и целостное хранение данных в NAND флеш-памяти путем реализации как базовых элементов FTL, таких как преобразование логических адресов (LBA) в физические адреса, сборщик мусора (Garbage Collection, GC), контроль равномерности износа NAND флеш-памяти (Wear-Leveling, WL) [17], управление поврежденными блоками NAND флеш-памяти, так и более сложных, связанных с алгоритмами коррекции ошибок в хранящихся данных и другие.

Помимо функциональных к современным флеш-накопителям предъявляются жесткие требования по характеристикам производительности, энергопотребления и QoS (Quality of Service). Обеспечение выполнения всех этих требований является достаточно сложной задачей, которая также решается программным обеспечением путем усложнения существующих алгорит-

мов, внедрения новых и балансировкой их совместной работы.

К флеш-накопителю могут предъявляться специфические требования, не оговоренные или не определенные стандартами, например, изменение поведения флеш-накопителя в разных температурных режимах его работы, сбор определенных статистических данных внутри флеш-накопителя с возможностью доступа к ним извне, особые требования по защите информации и ограничению доступа к данным накопителя. Все это также реализуется средствами программного обеспечения.

V. ВЕРИФИКАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ НАКОПИТЕЛЯ

Отдельная команда в SK hynix отвечает за программную верификацию – с использованием симулятора накопителя собственной разработки. Для верификации создаются модели (hardware models), позволяющие моделировать любое оборудование, внедрять ошибки (например, в NAND или RAM) и исполнять код на всех процессорах моделируемого накопителя или их подмножестве – их число может достигать до нескольких десятков для серверных SSD.

Одного симулятора и моделей недостаточно для масштабной верификации, также необходимы инструменты для отладки кода, необходимо автоматизировать управление тестовыми компаниями (это тысячи машин, на которых выполняется моделирование десятков тысяч тестов), также нужно как можно более упростить работу программиста, производящего анализ упавших тестов, т. к. при этом приходится иметь дело с огромным количеством кода – это сотни тысяч строк: встроенное ПО, симулятор, модели, тесты и т. д. Для этого разрабатывается ПО, позволяющее группировать возникшие проблемы и связывать эти группы с проблемами, выявленными ранее.

VI. ВАЛИДАЦИЯ ПРОГРАММНОЙ И АППАРАТНОЙ ЧАСТЕЙ НАКОПИТЕЛЯ

Одним из этапов разработки флеш-накопителя является валидация аппаратных модулей SnK, на базе которых построен накопитель. Данный процесс можно условно поделить на две части: валидация на FPGA и валидация непосредственно готового устройства (или его ревизий). На основе технических требований к SnK определяется и утверждается набор тестовых сценариев для всех аппаратных модулей. Эти сценарии реализуются в специализированном встроенном программном обеспечении. Тестовый фреймворк позволяет в автоматическом режиме запускать наборы тестов различных подсистем SnK. Выявленные дефекты на этапе валидации на FPGA исправляются проектировщиками SnK и перепроверяются последую-

щим регрессионным тестированием. Поскольку не весь аппаратный функционал может быть протестирован на FPGA, после изготовления СнК, производится тестирование непосредственно интегральной микросхемы. Это делается с помощью того же (или незначительно модифицированного) специализированного ПО. Дефекты, которые выявляются в готовых СнК исправляются в последующих его ревизиях. Кроме валидационного ПО для тестирования модулей и подсистем СнК используется различный аппаратный и программный инструментарий: логические и протокол-анализаторы, осциллографы, программно-аппаратные эмуляторы такие, как Velocce, тестовые фреймворки и системы как сторонних производителей, так и разрабатываемые внутри компании.

VII. ИССЛЕДОВАНИЯ В ОБЛАСТИ АППАРАТУРЫ КОНТРОЛЛЕРОВ NAND ФЛЕШ-ПАМЯТИ

В настоящее время в компании проводятся исследования различных структурных элементов контроллеров флеш-памяти. Результаты исследований, как правило, внедряются в продукты SK hynix, а также защищаются с помощью патентов США. Рассмотрим основные направления исследований.

Аппаратные ускорители. Существуют аппаратные решения на основе твердотельных накопителей, которые специально предназначены для хранения баз данных [19]. В связи с этим в компании были разработаны решения для аппаратной поддержки различных операций в системах управления базами данных, которые позволяют повысить производительность выполнения данных операций с незначительными аппаратными затратами (менее 1% общей аппаратуры контроллера). Также была предложена система управления памятью контроллера, которая позволяет динамически перераспределять нагрузку с одних типов памяти (например, статические ОЗУ) на другие (например, флеш-память) [20]. Для более детального анализа разрабатываемого аппаратного и программного обеспечения ускорителей были разработаны методы их отладки [21], позволяющие уменьшить зависимость от существующих дорогостоящих решений. В настоящее время одним из направлений исследований является разработка новых архитектур аппаратных ускорителей алгоритмов, реализованных на уровне встроеного ПО [22].

Безопасность накопителей информации на основе флеш-памяти. Проблемы безопасности твердотельных накопителей представляют интерес как для исследователей, так и для производителей данного класса устройств [23]. В соответствии с данным направлением в компании был разработан ряд аппаратных реше-

ний для уменьшения уязвимостей в различных структурных элементах контроллера накопителей информации на основе флеш-памяти. Генераторы случайных числовых последовательностей (ГСЧП) являются одним из важнейших элементов протоколов безопасности. В компании был разработан источник случайности общего назначения, основанный на асинхронном триггере и инверторе, а также генератор на базе чтения элементов флеш-памяти без коррекции ошибок [24]. В связи с необходимостью хранения части таблицы соответствия логических и физических адресов на стороне хоста в накопителях информации для мобильных устройств, был представлен метод кодирования и декодирования записей таблицы с помощью физически неклонированных функций (ФНФ) [25]. ФНФ были использованы в процессе улучшения аппаратуры для скремблирования данных, обладающей уязвимостями, которые приводят к более интенсивному износу накопителя информации [27]. Также был предложен аппаратный метод уникальной идентификации интегральных схем флеш-памяти, основанный на сравнении числа единиц при чтении различных страниц.

Надежность элементов флеш-памяти. Особенностью накопителей информации на основе NAND флеш является низкая надежность элементов памяти, уменьшающаяся с увеличением разрядности одной ячейки [28]. В настоящее время большинство накопителей на основе флеш-памяти построено на элементах, способных хранить три бита информации (Triple Layer Cell, TLC). Для улучшения надежности элементов памяти был предложен метод скремблирования, позволяющий сделать износ элементов более равномерным [26]. Также надежность элементов памяти обеспечивается путем сбора и анализа диагностической информации на уровне встроеного ПО. Основной проблемой сбора данной информации является многопроцессорность и гетерогенность контроллеров NAND флеш-памяти. Актуальным направлением в области повышения надежности накопителей является проектирование и разработка аппаратных методов самотестирования и диагностики элементов памяти и аппаратуры контроллера [29].

Анализ потока входных команд контроллера. Одним из актуальных направлений в развитии архитектуры контроллеров накопителей информации на основе флеш-памяти является анализ входного потока команд (Workload) [30]. В данном направлении был предложен алгоритм анализа и прогнозирования потока команд с целью оптимизации работы контроллера [31]. Также для решения проблемы фрагментации логических и физических адресов памяти был предложен алгоритм кластеризации входных команд [32]. Было представле-

но аппаратно-программное решение, основанное на компактном представлении входного потока команд. Предложенный подход основан на использовании рекуррентных автокодировочных нейронных сетей (Recurrent Autoencoder Neural Networks) и позволяет преобразовать поток команд в вектор, состоящий из вещественных чисел. Данное решение позволяет производить классификацию потоков входных команд с целью самотестирования устройства памяти, а также настройки параметров встроенного ПО для улучшения производительности и надежности. Перспективным направлением в данной области является внедрение методов машинного обучения для анализа входного потока команд с целью реконфигурации аппаратуры контроллера SSD.

Организация работы с патентными заявками. Компания SKHMS E с 2016 года подготовила 41 патентную заявку, 23 из которых были опубликованы в открытом доступе, а 18 – находятся на стадии оформления. По 8 заявкам уже получены патентные свидетельства. Процесс патентования технических решений состоит из нескольких этапов:

1. предварительное обсуждение патентной идеи авторов с экспертами;
2. разработка предварительной заявки, которая содержит краткое описание предметной области и основной идеи, а также детальное описание технической реализации;
3. обсуждение предварительной заявки в расширенной экспертной группе;
4. рецензирование заявки патентным бюро в головной компании SK hynix;
5. оформление патентной заявки юридическим бюро в США;
6. подача заявки в патентное бюро США (United States Patent and Trademark Office, USPTO);
7. опубликование заявки в открытом доступе;
8. экспертиза заявки патентным бюро США;
9. оформление патентного свидетельства.

Весь процесс от предложения идеи до получения патентного свидетельства может занимать от 2 до 5 лет.

В компании в процесс разработки патентных заявок вовлечены все департаменты, а также технические писатели. Успешные патентные заявки материально поощряются компанией.

VIII. Совместная учебная лаборатория в БГУИР

В сентябре 2018 года на факультете компьютерных систем и сетей БГУИР была открыта совместная учебная лаборатория SKHMS E [33]. Территориально лаборатория расположена в аудитории 402 корпуса №5. Возглавляет лабораторию профессор кафедры информатики Иванюк А.А.

Деятельность лаборатории нацелена на привлечение заинтересованных студентов, магистрантов и аспирантов в исследовательские и инженерные проекты в области проектирования контроллеров запоминающих устройств на основе NAND флеш-памяти. В первый год работы лаборатории ведущими специалистами SKHMS E были прочитаны открытые лекции по следующим направлениям: верификация аппаратуры контроллеров SSD, проектирование и отладка встроенного ПО контроллеров SSD, исследование и разработка специализированных алгоритмов, позволяющих улучшать основные характеристики SSD-накопителей, контроллеры SSD на основе микропроцессорных систем.

В 2019 году в лаборатории были прочитаны специализированные курсы по особенностям разработки ПО для моделирования, тестирования и верификации встроенного ПО, по архитектурным особенностям микропроцессоров и их применению в составе контроллеров SSD, теоретический курс по алгебраической теории помехоустойчивых кодов, применяемых в запоминающих устройствах на основе NAND флеш-памяти. Кроме этого проведены практические занятия по реализации базовых алгоритмов управления NAND флеш-памятью с применением отладочных плат на основе ARM-процессоров.

В начале 2020 года были проведены лекционные занятия по основам машинного обучения и возможности его применения в управлении NAND флеш-накопителями. В этом же году был реализован студенческий проект по сборке и запуску полноценного SSD-накопителя с применением платы быстрого прототипирования Cosmos/Cosmos+ FPGA Platform (см. рис. 3) [34]. Данная плата, основу которой составляет программируемая логическая интегральная схема Xilinx FPGA Zynq-7000, предназначена для совместного проектирования как аппаратной, так и программной части контроллера SSD. С целью вовлечения большего числа студентов в процесс проектирования контроллеров SSD были проведены лекции по основам технологий программируемых логических интегральных схем, HDL-языкам описания цифровой аппаратуры и системам ее автоматизированного проектирования, основам проектирования цифровых систем с использованием софт-процессоров. Для реализации студенческих проектов в лаборатории имеются платы быстрого прототипирования Digilent NEXYS4 [35] и Zybo Z7 [36], лицензионный пакет САПР Vivado/Vitis от Xilinx [37].

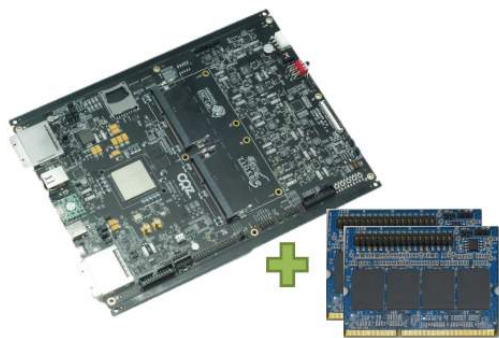


Рис. 3 – Плата быстрого прототипирования Cosmos/Cosmos+ FPGA Platform

В связи с пандемией COVID-19 учебная лаборатория перешла на индивидуальное взаимодействие со студентами, магистрантами и аспирантами. Каждый обучаемый в лаборатории вовлечен в академический исследовательский проект. По мере необходимости работы с оборудованием для слушателей выделен специальный график посещения лаборатории. Список тематических проектов разнообразен и включает в себя такие темы как:

- исследование конфигурируемых возможностей микропроцессорного ядра RISC-V;
- исследование и реализация алгоритмов скремблирования данных для SSD;
- исследование и аппаратно-программная реализация специализированных кодов коррекции пакетных ошибок;
- сравнение различных стратегий управления GC (Garbage Collection) для NAND флеш-памяти;
- исследование и прототипирование новых методов обеспечения защиты хранящихся данных, встроенного ПО и аппаратуры SSD-накопителей от несанкционированного доступа и использования.

Студенты совместной лаборатории учатся работать с англоязычной технической и научной литературой, работать над постановкой и проведением экспериментов, анализом полученных результатов. Помимо этого, студенты приобретают навыки участия в научно-технических конференциях, тренируются излагать свои мысли и результаты в форме тезисов докладов и при выступлениях на соответствующих секциях. Большинство обучаемых в лаборатории используют свои наработки в курсовом и дипломном проектировании, выбирают соответствующие исследовательские направления при обучении в магистратуре и аспирантуре.

В ближайшее время в лаборатории планируется открыть новое направление по верификации аппаратуры контроллеров SSD с изучением языка SystemVerilog и методологии UVM [38].

За время работы лаборатории в ней прошли обучения более 30 студентов с различных специальностей БГУИР. Пять из них уже успешно ра-

ботают на инженерных должностях в SKHMS E, двое из которых продолжают взаимодействие по своим академическим проектам, помогая другим студентам.

В лабораторию приглашаются студенты, магистранты и аспиранты, заинтересованные в изучении принципов функционирования современных запоминающих устройств на основе NAND флеш-памяти и технологий проектирования/верификации аппаратуры и встроенного программного обеспечения контроллеров SSD-дисков.

IX. ЗАКЛЮЧЕНИЕ

Разработка современных SSD-дисков является комплексной задачей, для успешного решения которой в SK Hynix развиваются сразу несколько научно-технических направлений: развитие технологии изготовления флеш-памяти, стандартизация процесса проектирования и верификации чипов SSD-контроллеров и встраиваемого ПО, непрерывное проведение исследовательской работы по всем связанным направлениям, взаимодействие и поддержка образовательных программ для подготовки высококвалифицированных инженеров.

X. СПИСОК ЛИТЕРАТУРЫ

1. Evolution of 4D NAND Flash Opens the Era of Terabyte Smartphones [Electronic resource] – Publ. date 25.09.2019. – Mode of access: <https://news.skhynix.com/evolution-of-4d-nand-flash-opens-the-era-of-terabyte-smartphones/>. – Date of access: 20.10.2021.
2. Park, J-W. A 176-Stacked 512 Gb 3b/Cell 3D-NAND Flash with 10.8 Gb/mm² Density with a Peripheral Circuit Under Cell Array Architecture / J-W. Park [et al.] // IEEE international solid-state circuits conference: proc. of the Intern. Conf., 13-22.02.2021.
3. SK hynix Gold P31 M.2 NVMe SSD Review: High-Performance, Unprecedented Efficiency [Electronic resource] – Publ. date 26.09.2021. – Mode of access: <https://www.tomshardware.com/reviews/sk-hynix-gold-p31-m2-nvme-ssd-review>. – Date of access: 20.10.2021.
4. NVM Express [Electronic resource] – Mode of access: <https://nvmexpress.org>. – Date of access: 20.10.2021.
5. JEDEC UFS & MIPI UniPro: Enabling Storage for Mobile, Automotive, and Other Applications [Electronic resource] – Mode of access: <https://www.mipi.org/sites/default/files/JEDEC-UFS-MIPI-UniPro-Webinar-Enabling-Storage-Mobile-Automotive.pdf>. – Date of access: 20.10.2021.
6. Rosenberg, S. A Practical Guide to Adopting the Universal Verification Methodology (UVM) / S. Rosenberg, K. Meade. – CA, USA : Cadence Design Systems, 2010. – 275 p.
7. Veloce HW-Assisted Verification System [Electronic resource] – Mode of access: <https://eda.sw.siemens.com/en-US/ic/veloce/>. – Date of access: 20.10.2021.
8. Emulation: High performance hardware and software verification and debug of complex SoCs and Systems [Electronic resource] – Mode of access: https://www.cadence.com/en_US/home/tools/system-design-and-verification/emulation-and-prototyping/palladium.html. – Date of access: 20.10.2021.

9. Emulation: Industry's Fastest Emulation Systems [Electronic resource] – Mode of access: <https://www.synopsys.com/verification/emulation.html>. – Date of access: 20.10.2021.
10. What Semifore Does [Electronic resource] – Mode of access: <https://semifore.com/about-semifore/>. – Date of access: 20.10.2021.
11. Confluence [Electronic resource] – Mode of access: <https://www.atlassian.com/software/confluence>. – Date of access: 20.10.2021.
12. MediaWiki: серебряная пуля или швейцарский нож? / С. Фомин // Открытые системы. СУБД, № 3, 2009 [Electronic resource] – Publ. date 30.04.2009. – Mode of access: <https://www.osp.ru/os/2009/03/8162918>. – Date of access: 20.10.2021.
13. Doxygen: Generate documentation from source code [Electronic resource] – Mode of access: <https://www.doxygen.nl/index.html>. – Date of access: 20.10.2021.
14. Natural Docs: Readable Source Code Documentation for 21 Programming Languages [Electronic resource] – Mode of access: <https://www.naturaldocs.org/>. – Date of access: 20.10.2021.
15. WaveDrom: Rendering Beautiful Waveforms from Plain Text / A. Chapyzenka, J. Probell [Electronic resource] – Mode of access: https://wavedrom.com/images/SNUG2016_WaveDrom.pdf. – Date of access: 20.10.2021.
16. Jenkins [Electronic resource] – Mode of access: <https://www.jenkins.io>. – Date of access: 20.10.2021.
17. Ming-Chang, Y. Garbage collection and wear leveling for flash memory: Past and futur / Y. Ming-Chang [et al.] : proc. of the 2014 International Conference on Smart Computing / SMARTCOMP. – 2015. – Pp. 66-73.
18. Гибридная верификация процессоров Baikal: косимуляция с FPGA-платформой прототипирования Synopsys HAPS-80 [Электронный ресурс] / «Навр». – Москва, 2021. – Режим доступа: <https://habr.com/ru/company/baikalelectron/blog/582782/>. – Дата доступа: 20.10.2021.
19. Park, J.-H. SaS: SSD as SQL Database System / J.-H. Park, S. Choi, G. Oh, S.-W. Lee // Proc. VLDB Endowment. – 2021. – Vol. 14, № 9. – pp. 1481–1488.
20. Microprocessor-based system memory manager hardware accelerator [Electronic resource] : pat. app. US20210255956A1 / A. Kuyel. – Publ. date 19.08.2021. – Mode of access: <https://patents.google.com/patent/US20210255956A1>. – Date of access: 20.10.2021.
21. Method of debugging hardware and firmware of data storage [Electronic resource] : pat. US10970442B1 / A. Kuyel. – Publ. date 06.04.2021. – Mode of access: <https://patents.google.com/patent/US10970442B1>.
22. Dally, W. J. Domain-Specific Hardware Accelerators / W. J. Dally, Y. Turakhia, S. Han // Comm. ACM. – 2020. – Vol. 63 № 7. – pp. 48–57.
23. Hassell, J. The importance of SSD security and data encryption [Electronic resource]. – Mode of access: <https://insights.samsung.com/2021/06/10/the-importance-of-ssd-security-and-data-encryption/>. – Samsung, 2021. – Date of access: 20.10.2021.
24. Raw read based physically unclonable function for flash memory [Electronic resource] : pat. app. US20210055912A1 / S. S. Zalivaka, A. A. Ivaniuk – Publ. date 25.02.2021. – Mode of access: <https://patents.google.com/patent/US20210055912A1>.
25. Encoder and decoder using physically unclonable functions [Electronic resource] : pat. app. US20210226772A1 / S. S. Zalivaka, A. A. Ivaniuk – Publ. date 22.07.2021. – Mode of access: <https://patents.google.com/patent/US20210226772A1>.
26. Data scrambler for memory systems and method thereof [Electronic resource] : pat. app. US20210303715A1 / S. S. Zalivaka, A. A. Ivaniuk – Publ. date 30.09.2021. – Mode of access: <https://patents.google.com/patent/US20210303715A1>.
27. Data scramblers with enhanced physical security [Electronic resource] : pat. app. US20210326490A1 / S. S. Zalivaka, A. A. Ivaniuk – Publ. date 21.10.2021. – Mode of access: <https://patents.google.com/patent/US20210326490A1>.
28. Cai, Y. Vulnerabilities in MLC NAND Flash Memory Programming: Experimental Analysis, Exploits, and Mitigation Techniques / Y. Cai [et al.] // IEEE Int. Symp. High Perf. Comp. Arch. (HPCA'17) – Austin, USA, 2017. – pp. 49–60.
29. Lai, K. H. Why Next-Generation NAND Flash Requires a Dedicated Test Solution [Electronic resource]. – Mode of access: <https://www3.advantest.com/documents/11348/11d6ba1a-7d3f-4f66-acf3-cd9d1167f924>. – Advantest, 2013. – Date of access: 20.10.2021.
30. Seo, B. IO Workload Characterization Revisited: A Data-Mining Approach / B. Seo [et al.] // IEEE Trans. on Comp. – 2014. – Vol. 63 № 13. – pp. 3026–3038.
31. Workload prediction in memory system and method thereof [Electronic resource] : pat. US10802761B2 / A. Charnevich, S. S. Zalivaka – Publ. date 13.10.2020. – Mode of access: <https://patents.google.com/patent/US10802761B2>.
32. Workload clusterization for memory system and method of executing the same [Electronic resource] : pat. app. US20190361628A1 / Y. Yaromenka, A. Charnevich, J. M. Koo, S. S. Zalivaka – Publ. date 28.11.2019. – Mode of access: <https://patents.google.com/patent/US20190361628A1>.
33. В ВГУИР открылась новая лаборатория в партнерстве с IT-компанией SK hynix memory solutions Eastern Europe [Электронный ресурс]. – Режим доступа: <https://www.bsuir.by/ru/news/101971-v-bguir-otkrylas-novaya-laboratoriya-v-partnerstve-s-it-kompaniy-sk-hynix>. – Дата доступа: 20.10.2021.
34. OPEN-SOURCE SOLID-STATE DRIVE PROJECT FOR RESEARCH AND EDUCATION [Electronic resource]. – Mode of access: <http://openssd.io/>. – Date of access: 20.10.2021.
35. Nexys 4 Artix-7 FPGA Trainer Board [Electronic resource]. – Mode of access: <https://digilent.com/shop/nexys-4-artix-7-fpga-trainer-board/>. – Date of access: 20.10.2021.
36. Zybo Z7: Zynq-7000 ARM/FPGA SoC Development Board [Electronic resource]. – Mode of access: <https://digilent.com/shop/zybo-z7-zynq-7000-arm-fpga-soc-development-board/>. – Date of access: 20.10.2021.
37. Vivado ML Overview [Electronic resource]. – Mode of access: <https://www.xilinx.com/products/design-tools/vivado.html>. – Date of access: 20.10.2021.
38. UVM Verification Primer [Electronic resource]. – Mode of access: <https://www.doulos.com/knowhow/systemverilog/uvm/uvm-verification-primer/>. – Date of access: 20.10.2021.

ОНТОЛОГИЧЕСКИЙ ПОДХОД К КОМПЛЕКСНОЙ АВТОМАТИЗАЦИИ ПРЕДПРИЯТИЙ РЕЦЕПТУРНОГО ПРОИЗВОДСТВА В КОНТЕКСТЕ КОНЦЕПЦИИ INDUSTRY 4.0

Иванюк Д. С., Пупена А. Н., Шункевич Д. В.
ОАО «Савушкин продукт»

Национальный университет пищевых технологий
Белорусский государственный университет информатики и радиоэлектроники
Брест, Республика Беларусь
Киев, Украина

Минск, Республика Беларусь

E-mail: id@pda.savushkin.by, pupena_san@ukr.net, shunkevich@bsuir.by

В работе рассматривается онтологический подход к проектированию предприятий рецептурного производства, предполагающий переход от рассмотрения производственных процессов в рамках одного предприятия к рассмотрению процессов, охватывающих, в соответствии с концепцией Industry 4.0, и смежные предприятия – поставщиков сырья (молочные фермы) и оптовых потребителей продукции (магазины, торговые сети). Рассмотрены примеры реализации указанного подхода на предприятии ОАО «Савушкин продукт», в частности, рассматривается описание моделей предприятия на основе онтологического подхода, реализация подсистемы информационного обслуживания сотрудников в рамках SCADA-системы, подсистема управления логистическими процессами. Кроме того, рассматривается агентно-ориентированный подход к организации взаимодействия роботов в рамках роботизированных производственных комплексов, основанный на взаимодействии через общую семантическую память.

ВВЕДЕНИЕ

Данная статья рассматривает дальнейшее развитие автоматизации предприятий рецептурного производства на примере ОАО «Савушкин продукт». Оно состоит в переходе от рассмотрения производственных процессов, происходящих в рамках конкретного предприятия, к рассмотрению полного продуктового цикла – от получения заявки от магазина-клиента до ее выполнения (удовлетворения запроса потребителя). Данный процесс, кроме самого предприятия, включает в качестве субъектов молочные фермы, логистические службы, магазины и др. Также даже в рамках одного предприятия на сегодняшний день используются разрозненные системы (несколько SCADA-систем, система управления транспортом, система управления складом и др.). Стыковка таких систем и поддержка согласованности информации в них осуществляется вручную (либо опускается). Сейчас существует потребность автоматизировать согласование информации и обеспечить, там где это уместно, взаимодействие устройств на разных этапах продуктового цикла. Наличие такой потребности привело к появлению инициативы Industry 4.0 в Германии и ей аналогичных в других странах мира.

Целью работы является развитие моделей, рассмотренных в работах [1] [2] [3], в соответствии с концепцией Industry 4.0.

I. КОНЦЕПЦИЯ INDUSTRY 4.0 И КИБЕРФИЗИЧЕСКИЕ СИСТЕМЫ

Концепция Industry 4.0 (Industrie 4.0) была сформулирована в Германии в 2011 году. Она

подразумевает создание и внедрение в производство т.н. киберфизических систем (КФС) и использование интернета вещей и услуг в производственных процессах [4]. Сейчас под термином Industry 4.0 подразумевают стек новых технологий и принципов для управления современным производством во всем мире. Многие страны разрабатывают свои стратегии и дорожные карты под различными названиями: «умная фабрика» (Нидерланды), «фабрика будущего» (Италия), «промышленный интернет вещей» (США) и другие.

Основные принципы концепции Industry 4.0 [5]:

- **Взаимодействие.** Возможность взаимодействия устройств, датчиков, людей посредством Интернета вещей (IoT), Интернета людей (IoP), Интернета услуг (IoS).
- **Виртуализация.** Означает способность киберфизической системы контролировать физические процессы. Данные сенсоров проецируются на модель предприятия, включающую состояние всех киберфизических систем. В случае возникновения нештатной ситуации должна быть возможность уведомить оператора, предоставив ему информацию по ее устранению и обеспечению безопасности, тем самым осуществляя поддержку принятия решений персоналом.
- **Децентрализация.** Растущая потребность в штучных партиях заказных продуктов увеличивает сложность централизованного управления производством. КФС

могут иметь встроенные вычислительные модули, позволяющие им принимать решения самостоятельно и переадресовывать задачу управляющей системе только в случае необходимости. Несмотря на это, необходимо обеспечить контроль качества конечного продукта и прослеживаемость, что требует централизованного управления. К примеру, необходимые шаги производственного процесса могут быть закодированы в RFID-метках, что освобождает от необходимости централизованного управления данным аспектом производства малых партий продукта.

- **Анализ и реагирование в реальном времени.** Для управления производством необходимо, чтобы данные с сенсоров постоянно собирались и анализировались в режиме реального времени. В случае отказа одной производственной установки можно «перепоручить» ее задачу другой.
- **Ориентированность на услуги.** Услуги компаний, КФС и людей доступны в Интернете услуг и могут быть использованы другими участниками. Услуги могут предоставляться как внутри предприятия, так и другим предприятиям. КФС предоставляют свои услуги в виде веб-служб. Это позволит реализовать производство продукта путем комбинирования производственных операций в соответствии со спецификацией клиента, закодированной, например, на RFID-метке.
- **Модульность.** Система должна быть гибкой, т.е. легко адаптируемой к меняющимся требованиям (например, сезонным изменениям в потреблении, изменению характеристик продукта или производства). Адаптация должна осуществляться заменой или расширением отдельных компонентов системы. Обеспечение совместимости компонентов требует наличия стандартизированных механизмов взаимодействия, позволяющих автоматически идентифицировать компоненты и включать их в Интернет услуг.

Под КФС понимается совокупность интеллектуальных, легко интегрируемых физических компонентов со встроенными в них вычислительными ресурсами, тесно взаимодействующих между собой и отслеживающих изменения в состоянии внешнего мира [6].

Для построения КФС предприятия необходима интеграция вычислительных ресурсов и физико-технических процессов. Сенсоры, манипуляторы, информационные и управляющие системы должны взаимодействовать между собой на всех этапах производства, в том числе и выходящих за рамки одного конкретного предприятия [7]. Следует, однако, понимать, что внед-

рение новых систем автоматизации, в том числе интеллектуальных, на предприятии должно основываться на уже имеющихся у предприятия ресурсах [8].

Потребность в комплексной автоматизации сложных процессов, требующих согласованной работы множества служб и технических средств, создает потребность в разработке подобных систем. В дальнейшем будем рассматривать повышение уровня автоматизации предприятий рецептурного производства в рамках проектирования КФС, отвечающей за производство творога «Хуторок» на ОАО «Савушкин продукт».

Такая КФС должна обеспечивать комплексную информационную поддержку и автоматизацию процесса производства творога «Хуторок» от доения коровы до поступления готовой продукции в магазин. В этом процессе можно выделить следующие основные стадии:

- сбор молока на ферме;
- доставка молока с фермы на молокозавод;
- обработка молока на молокозаводе и получение творога;
- формовка и упаковка творога;
- доставка продукции со склада завода на склад заказчика (магазина, торговой сети и т.п.).

II. ПРОБЛЕМЫ РАЗРАБОТКИ СИСТЕМ КОМПЛЕКСНОЙ АВТОМАТИЗАЦИИ ПРЕДПРИЯТИЯ И ПРЕДЛАГАЕМЫЙ ПОДХОД К ИХ РЕШЕНИЮ

Основной проблемой, возникающей при разработке системы комплексной автоматизации, является проблема интеграции различных компонентов системы и обеспечения взаимодействия между ними. Решение данной проблемы традиционным способом, предполагающим разработку средств коммуникации между разнородными компонентами системы (интерфейсов, протоколов и т.д.), с одной стороны, приводит к существенным накладным расходам, связанным с разработкой таких средств, с другой стороны, сильно усложняет архитектуру такой системы, значительно повышая трудоемкость ее сопровождения и развития. Очевидно, что постоянное развитие технологий производства на различных его этапах и расширение самого производства обуславливает требование гибкости к системе автоматизации, то есть способности легко дополнять ее различными компонентами и при необходимости модифицировать уже существующие компоненты.

Для решения перечисленных проблем предлагается развитие идей онтологического подхода к проектированию предприятий на основе Технологии OSTIS [1] [2]. Предприятие предлагается рассматривать как единую информационную многоагентную систему, в рамках которой:

- вся информация объединяется в единое информационное пространство (базу знаний предприятия, которая хранится в семантической памяти);
- все участники процесса (люди, роботизированные системы, различного рода производственные комплексы и т.д.) трактуются как агенты над этой общей базой знаний. Это означает, что они (а) отслеживают интересующие их ситуации в базе знаний и реагируют на них, (б) описывают результаты своей деятельности в базе знаний для того, чтобы эта информация была доступна другим агентам и они могли ее анализировать. В конечном итоге весь менеджмент процессов производства при таком подходе сводится к грамотной спецификации задач в такой общей базе знаний, указанию их приоритетов, исполнителей, сроков выполнения и т.д.;
- база знаний системы имеет иерархическую организацию, т.е. представляет собой иерархию предметных областей и соответствующих им онтологий;
- сама многоагентная система также имеет иерархическую организацию: агенты могут объединяться в коллективы произвольной степени вложенности (членом коллектива агентов также может быть коллектив агентов). Например, группа роботизированных систем может быть логически (или даже физически) объединена в целый роботизированный комплекс, способный решать задачи определенного класса.

Многоагентная система над общей базой знаний, реализованная по Технологии OSTIS, таким образом, отвечает концепции Industry 4.0 и, следовательно, может рассматриваться как киберфизическая:

- общая база знаний реализует принцип взаимодействия людей, датчиков и устройств, выступая посредником такого взаимодействия, виртуализации – в базе знаний представляется модель предприятия и сопутствующих физических процессов с необходимым уровнем детализации, ориентированности на услуги, – каждый участник производственного процесса (агент) специфицируется в рамках общей базы знаний, включая свою функциональность (предоставляемые услуги), модульности – за счет представляемых в ней библиотек совместимых, повторно используемых компонентов.
- многоагентный подход реализует принцип децентрализации по определению многоагентной системы [9], анализа и реагирования – агенты отслеживают состояние базы знаний предприятия и активизируются при возникновении определенных ситуаций (в том числе нештатных).

Предложенный подход обладает рядом преимуществ, таких как:

- отсутствие необходимости разработки средств непосредственного взаимодействия компонентов системы (человек-роботизированная система, человек-человек и т.д.) за счет их взаимодействия посредством общей памяти;
- за счет того, что все агенты взаимодействуют посредством общей памяти, в общем случае для системы не важно, как физически устроен тот или иной агент. Таким образом, постепенная замена ручного труда автоматизированными системами или совершенствование таких систем не требует внесения изменений в общую систему автоматизации;
- за счет использования общей единой базы знаний и широких возможностей ассоциативного поиска в такой базе знаний любой участник процесса производства в любой момент имеет доступ ко всей необходимой ему информации, а не к каким-либо заранее предусмотренным ее фрагментам, расширение числа которых может быть связано с дополнительными накладными расходами. Таким образом, существенно облегчается процесс мониторинга различных процессов и ускоряется поиск ответов на интересующие пользователя вопросы. При этом запросы пользователя к системе могут уточняться различными способами;
- одна и та же информация, хранимая в базе знаний, может по-разному визуализироваться для различных категорий пользователей, при этом сама информация будет оставаться неизменной – будут меняться только средства ее отображения. Таким образом, отсутствует необходимость дублирования информации;
- поскольку все производственные процессы специфицируются в базе знаний и управляются ей, внесение изменений в такие процессы в общем случае сводится к внесению изменений в базу знаний и при необходимости – к замене соответствующего оборудования. При этом существенно снижаются накладные расходы на перепрограммирование компонентов системы, налаживание взаимодействия между ними;
- спецификация всех производственных процессов в единой базе знаний предоставляет широкие возможности для их автоматического анализа, в том числе – постоянного мониторинга текущих процессов, автоматического выявления и устранения нештатных ситуаций, оптимизации текущих процессов, автоматического планирования будущих процессов и т.д.

III. АРХИТЕКТУРА ПРЕДЛАГАЕМОЙ СИСТЕМЫ

Разработка системы комплексной автоматизации предполагает выделение нескольких подсистем, ориентированных на решение задач определенных классов. В настоящее время практически на любом предприятии некоторые из этих подсистем уже реализованы с использованием современных технологий и выполняют свои классические функции.

К основным производственным подсистемам относятся (от более низкого уровня к более высокому):

- SCADA – решается задача диспетчерского управления технологическим процессом (оператором), сбора и архивации технологических данных;
- MES (управление производственными процессами) – осуществляется выпуск продукции в рамках производства;
- WMS (управление складом) – осуществляется хранение готовой продукции до отгрузки заказчику;
- ERP (планирование ресурсов предприятия) – определяет, какой продукт в каких количествах и на каких производственных мощностях будет производиться.

При этом каждый уровень связан не только с последующим, но и с предыдущим. Таким образом, одной из задач при построении и внедрении системы комплексной автоматизации является интеграция уже существующих подсистем в единую систему и обеспечение возможности развития указанных систем без разрушения единой комплексной системы производства. В конечном итоге построение такой системы предполагает переход от интеграции разнородных подсистем к единому унифицированному технологическому фундаменту для реализации всех подсистем перечисленных классов.

Предлагаемая система разрабатывается на основе Технологии OSTIS, а значит, в соответствии с ней состоит из базы знаний, машины обработки знаний и пользовательского интерфейса.

В общем случае база знаний системы комплексной автоматизации предприятия содержит [1]:

- онтологии соответствующих отрасли промышленных стандартов, таких как ISA-88 [10];
- модели предприятия, построенные на основе этих онтологий (в случае ISA-88 – физическая, процедурная, процессная и др.);
- онтологии усовершенствования предприятия, формализующие принципы усовершенствования и адаптации предприятия к меняющимся внешним условиям;
- средства коллективной разработки баз знаний предприятия и соответствующих компонентов машин обработки знаний;

- модели пользовательских интерфейсов систем управления предприятием;
- модели информационного обслуживания различных категорий пользователей;
- модели представления знаний предприятия, с помощью которых предприятие можно описать во всех необходимых аспектах:
 - модель управления знаниями предприятия [11];
 - онтологическая модель предприятия [12];
 - многоагентная модель предприятия [9];
 - модель ситуационного управления предприятием [13];
 - модель реинжиниринга бизнес-процессов [14].

Кроме того, для комплексного описания процесса производства база знаний системы комплексной автоматизации предприятия должна, среди прочих, содержать следующие модели:

- модели языков описания ячеек производства;
- модели ячеек производства, описанные с помощью этих языков;
- модели логистических процессов;
- модель взаимодействия промышленных роботов.

Для описания моделей ячеек производства на предприятиях рецептурного производства используется, в частности, стандарт ISA-88. В работе [1] рассмотрена формализация стандарта ISA-88 в виде семейства онтологий, показаны фрагменты моделей предприятия, построенных на его основе. В данной работе будет рассмотрен язык спецификации процедурных моделей предприятия (PFC), который описан в главе 6 второй части данного документа [15].

Модель логистических процессов необходима на всех этапах производства для описания внутренних (межцеховая логистика, складская логистика) и внешних логистических процессов (взаимодействие с фермами – поставщиками сырья – и магазинами – заказчиками конечной продукции). Модели языков описания и описанные с их помощью модели ячеек используются в рамках третьей и четвертой стадий, упомянутых во введении к статье, для формализации процесса производства и структуры используемого для этого оборудования.

Модель взаимодействия промышленных роботов дополняет физическую модель предприятия, причем понятие робота имеет расширенную трактовку: под ним могут пониматься также технологические установки и их комплексы, работающие либо с минимальным вмешательством человека, либо вовсе без него.

Далее рассмотрим конкретные примеры реализации предложенных ранее принципов в рам-

ках различных подсистем разрабатываемой системы.

IV. ОПИСАНИЕ МОДЕЛЕЙ ПРОИЗВОДСТВЕННЫХ ПРОЦЕССОВ НА ОСНОВЕ ОНТОЛОГИЧЕСКОГО ПОДХОДА

Для удобства работы персонала предприятия с системой комплексной автоматизации необходимо обеспечить, с одной стороны, возможность взаимодействия системы с пользователями удобными им способами (в том числе – с использованием различных графических языков, на ограниченном естественном языке и посредством речевых сообщений), с другой стороны, обеспечить возможность добавления в систему новых языковых средств, например, новых графических языков.

Каждая интеллектуальная система оперирует базой знаний на внутреннем языке, и диалог осуществляется как обмен сообщениями между пользователем и системой. Чтобы такой диалог состоялся, необходимо отобразить фрагмент базы знаний во внешнюю форму. Такая форма может иметь как универсальный, так и специализированный характер. Трансляция с внутреннего языка на внешний и обратно в рамках Технологии OSTIS организовывается так, чтобы механизмы трансляции не зависели от внешнего языка; для реализации нового специализированного языка в таком случае необходимо будет только описать его синтаксис и семантику, универсальная же модель трансляции не будет зависеть от данного описания.

Механизм трансляции обеспечивается благодаря наличию в системе набора рецепторных и эффекторных агентов [16] в режиме постоянного обмена сообщениями между пользователем и системой. Данный механизм обмена сообщениями выглядит следующим образом:

- пользователь с помощью редактора на синтаксисе одного из специализированных внешних языков записывает некоторую информацию;
- рецепторные агенты фиксируют факт начала трансляции записанной синтаксической структуры;
- эффекторные агенты преобразуют синтаксическую структуру с помощью набора правил в последовательность элементов, составляющих фрагмент семантически связного sc-текста, однозначно интерпретируемого в памяти системы;
- если пользователь делает изменения в полученном sc-тексте, то происходит обратный процесс: устанавливается соответствие между знаками сущностей и их изображениями, в результате чего отображается синтаксическая структура на выбранном языке.

Семантика любого языка подразумевает выделение набора однозначно определенных сущностей, достаточных для написания текстов, несущих смысловую нагрузку для пользователя или машины (системы). Под текстами языка понимаются синтаксические структуры, являющиеся изображениями сущностей языка. Накопленные правила языка задают однозначные соответствия между множеством сущностей (алфавитом языка) и множеством изображений (знаков файлов), используемых для трансляции текстов в память интеллектуальной системы и из нее, а также для визуализации этих текстов.

В процессе описания каждого языка можно выделить некоторые аспекты, являющиеся общими для всех языков или для конкретного семейства языков. Исследования в этой области направлены на обоснование некоторого метаязыка, задающего структуру описания большинства существующих языков. Данный метаязык даст толчок к развитию естественно-языковых интерфейсов и позволит привести в лингвистические аспекты любого языка алгоритмическую точность.

Технология производства творога может быть описана по стандарту ISA S88. В контексте автоматизированного производства и стандарта ISA-88 выделяются следующие специализированные внешние языки: язык описания процедурной модели (PFC) и язык описания физической модели (P&ID), который, впрочем, не является частью стандарта ISA-88, но описывает оборудование на самом нижнем уровне иерархии согласно стандарту ISA-5.1 и аналогичным. С помощью языка PFC будет описан фрагмент производственной ячейки по изготовлению творога «Хуторок», который будет рассмотрен далее.

Язык PFC определяется в главе 6 стандарта ISA-88.00.02 и предназначен для описания рецептов со сложными процедурами, включающими параллельные шаги и условные ветвления. PFC-диаграммы изображают процедурную логику, используя набор значков, соединенных направленными связями, указывающими порядок исполнения процедурных элементов.

Алфавит языка PFC включает следующие элементы:

- процедурные элементы – элементы процедурной иерархии (фазы, операции и др.);
- дополнительные элементы – элементы, отвечающие за распределение, согласование и движение продукта в рамках процедуры (элемент размещения, элемент синхронизации);
- структуры, представленные в виде классов временных сущностей, задающих порядок исполнения процедурных элементов (выбор ветви исполнения, параллельно исполняемые ветви);

- переходы между процедурными и дополнительными элементами (неявный и явный переход, требующий выполнения некоторого условия).

В качестве примера разработки процедурной модели будет использована ячейка производства творога «Хуторок» предприятия «Савушкин продукт». Данная ячейка отражает стадию обработки молока на заводе на примере изготовления конкретного продукта. На рисунке 1 показаны коагуляторы, используемые для приготовления творога:



Рис. 1 – Коагуляторы

Мастер-рецепт описывает процесс от обработки молочной смеси до фасовки творожной массы. Рецептурная и аппаратная процедуры фокусируются на изготовлении творожной массы как на этапе выполнения мастер-рецепта. Наконец в рамках аппаратной процедуры акцентируется внимание на операции отделения и откачки сыворотки.

Мастер-рецепт изготовления творога «Хуторок» включает в себя следующие этапы:

- Подогрев основы до 28 градусов по Цельсию с параллельным внесением закваски;
- Заквашивание;
- Сквашивание;
- Нагрев;
- Разрезка;
- Отделение сыворотки;
- Откачка сыворотки с параллельным вымешиванием при необходимости;
- Подготовка к выдаче.

Запись аппаратной процедуры на языке PFC представлена на рисунке 2.



Рис. 2 – Аппаратная процедура производства творога

SCg-представление аппаратной процедуры производства творога (Рис. 3)

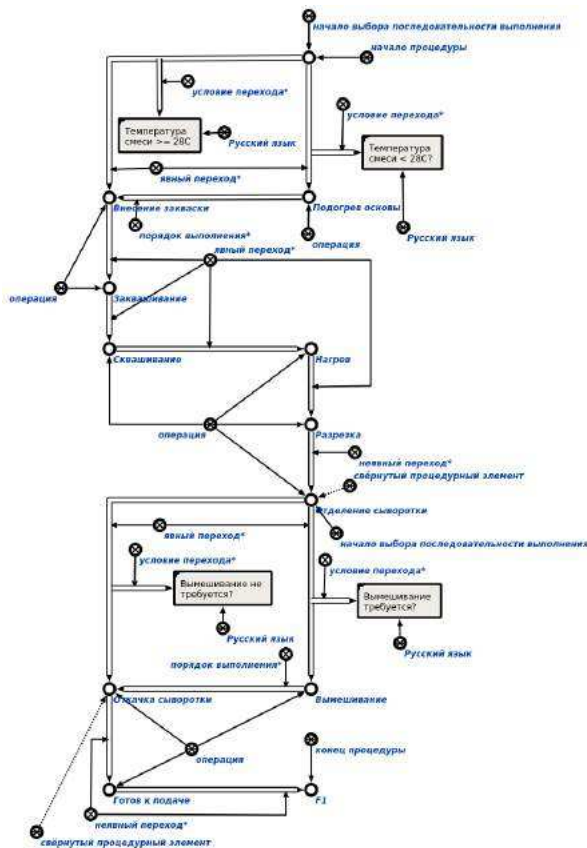


Рис. 3 – SCg-представление аппаратной процедуры производства творога

V. РЕАЛИЗАЦИЯ SCADA-ПОДСИСТЕМЫ

В рамках SCADA-подсистемы на данный момент реализована подсистема информационного обслуживания сотрудников. Основной задачей данной SCADA-системы, связанной с текущей реализацией, является предоставление в интерактивном режиме различной справочной информации об объектах и понятиях, используемых в SCADA-системе. Данная подсистема, в первую очередь, ориентирована на обслуживание мастера и начальника цеха.

В текущей реализации принцип работы системы заключается в следующем: в рамках интерфейса SCADA-системы присутствуют интерактивные элементы, которым однозначно соответствуют объекты и понятия в базе знаний системы информационного обслуживания (в настоящее время связь устанавливается через основную русскоязычный идентификатор [16]). При взаимодействии пользователя с интерактивным элементом в рамках SCADA-системы отправляется запрос к системе информационного обслуживания, содержащий идентификатор запрашиваемого элемента, после чего система отображает семантическую окрестность запрашиваемого элемента в текущем состоянии базы знаний.

В качестве примера рассмотрим фрагмент SCADA-системы «Хуторок». Пользователь (мастер) хочет получить дополнительную справочную информацию о текущем управляющем ре-

цепте. Для этого он нажимает на соответствующую кнопку в проекте (Рис. 4), после чего в браузере отображается ответ на запрос семантической окрестности понятия «управляющий рецепт» на языке SCn (Рис. 5).

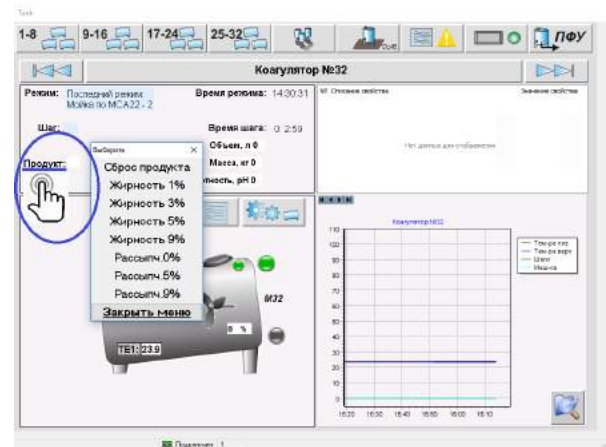


Рис. 4 – Запрос в SCADA-системе

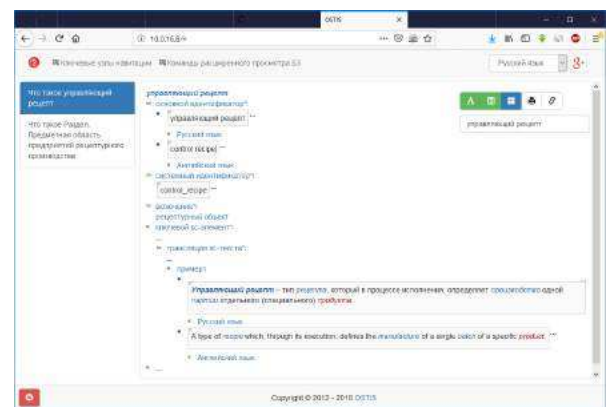


Рис. 5 – Ответ системы информационного обслуживания

Другой вариант использования системы информационного обслуживания мастером заключается в выявлении причин сложившейся ситуации (как штатной, так и нештатной). В текущей версии системы для решения данной задачи в составе решателя задач системы информационного обслуживания реализован *Абстрактный sc-агент поиска причин текущего состояния заданного объекта*. Указанный sc-агент находит в базе знаний все действия, в результате выполнения которых было изменено состояние объекта, являющегося аргументом запроса.

Пример работы данного sc-агента приведен на рисунках 6 и 7. Клапан K1 Valve2 является частью коагулятора K1 и в настоящий момент открыт (Рис. 6). После задания вопроса система в результате работы sc-агента выдает ответ о том, что клапан открыт, поскольку в данный момент для коагулятора K1 выполняется операция мойки (Рис. 7).

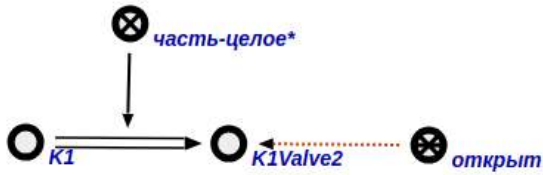


Рис. 6 – Связь клапана и коагулятора

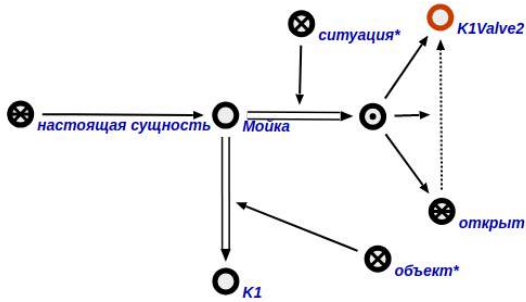


Рис. 7 – Причина открытия клапана

VI. РЕАЛИЗАЦИЯ ПОДСИСТЕМЫ ПО ЛОГИСТИКЕ

Основной задачей подсистем по логистике является обеспечение эффективного взаимодействия собственно производства, склада и транспорта.

В частности, логистическая цепочка производства творога «Хуторок» включает следующие этапы (Рис. 8)

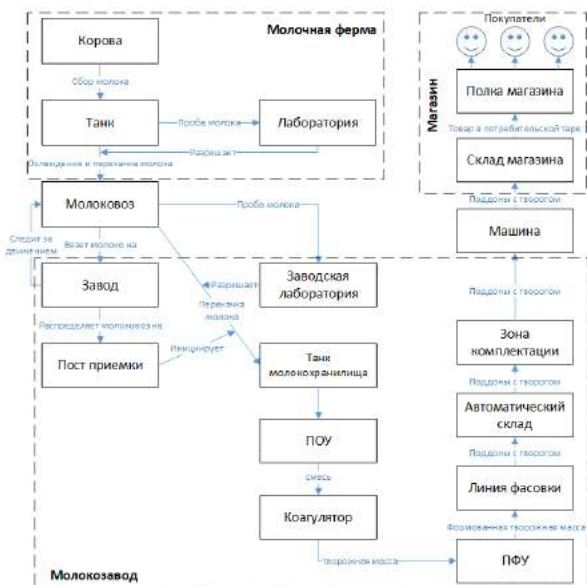


Рис. 8 – Логистическая цепочка производства творога

Любое ожидание и простои (ожидание приготовления, охлаждение продукта и простои машины во время ожидания загрузки) увеличивают себестоимость продукции. Таким образом,

для минимизации расходов необходимо, с одной стороны, обеспечить минимальную задержку между указанными этапами, с другой стороны, при возникновении задержек оперативно изменять время начала следующих этапов.

В свою очередь, перечисленные этапы могут делиться на более простые. Например, приготовление партии заключается в выполнении одного или нескольких рецептов на производственных аппаратах (в соответствии с S88), а рецепты состоят из последовательно выполняемых операций. Если операция во время выполнения по каким-либо причинам становится на паузу, то это означает, что время приготовления рецепта увеличивается. Сведения о данном событии должны поступать в базу знаний системы информационного обслуживания, после чего могут быть использованы для автоматической или ручной оценки критичности ситуации и, при необходимости, корректировки дальнейших этапов.

Например, если задержка в процессе приготовления партии превышает определенную величину, то необходимо отложить время загрузки данной партии для доставки заказчику; в противном случае машина приедет и будет ожидать, пока продукт охладится на складе. Система информационного обслуживания может отслеживать такие ситуации и в случае необходимости изменять время выезда машины на склад для загрузки с учетом нового времени готовности партии.

Пример правила в базе знаний, описывающего такого рода корректировку, представлен на рисунке 9 (предполагается, что задержка не должна превышать 30 мин.).

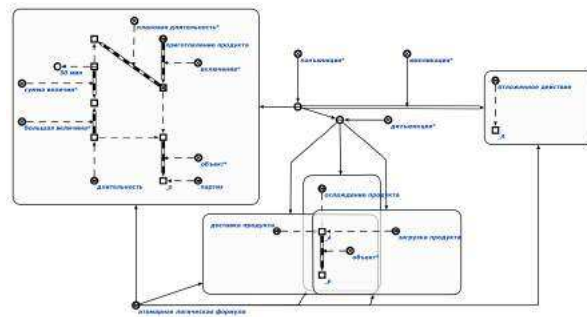


Рис. 9 – Правило, описывающее корректировку в случае задержки на производстве

VII. РЕАЛИЗАЦИЯ РОБОТОТЕХНИЧЕСКОЙ ПОДСИСТЕМЫ

Важнейшим компонентом системы комплексной автоматизации является подсистема, ориентированная на работу с роботизированными установками. Согласно изложенным выше принципам построения системы автоматизации предприятия, все участники процесса производства, включая роботов, тракуются как агенты, работающие над общим информационным пространством. Таким образом, с одной стороны, ро-

боты фиксируют результаты своей деятельности в общей базе знаний, с другой стороны, могут управляться ситуациями и событиями в базе знаний. Рассмотрим более подробно использование робототехнической подсистемы на примере линии производства творога.

Линия производства творога на данный момент включает в себя три роботизированных узла: узел упаковки продукта в короба, узел паллетирования и узел транспортировки паллет с готовым продуктом в приемный порт склада на основе мобильного робота. Первые два используют коллаборативного робота.

Рассмотрим задачу упаковки творогов в короба коллаборативным роботом. Общий вид установки приведен на рисунке 10.



Рис. 10 – Установка по укладке творогов

Установка получает на вход поток творогов, который затем разделяется и формируется в группы по 12 шт. Робот с помощью захватного механизма забирает партию и укладывает в две коробки. Упаковка творога допускается в два типа коробок: одноуровневые по 6 шт. и в двухуровневые – по 12. Далее коробки по выходному транспортеру выдаются на узел паллетизации. Для работы системы роботу необходима следующая информация, получаемая от системы информационного обслуживания:

- текущая производительность линии;
- тип подаваемой коробки;
- отсутствие коробок на входе;
- состояние соседних с роботом узлов производственной линии (работает, не работает, неисправен, авария);
- время планового обслуживания захватного механизма.

На основе данной информации программное обеспечение робота гибко реагирует на текущие параметры всего производственного процесса. Подстраивая скорость и задержки в выполнении операций, робот динамически управляет производительностью ячейки, чтобы не создавать очереди на входном и выходном конвейерах. Тип подаваемой коробки определяет количество продукта, которое роботу необходимо уложить в

нее. Состояние соседних узлов производственного процесса позволяет роботу переключить процесс упаковки в режим ручной обработки или пустить продукт во временный накопитель.

Следующий узел – узел паллетирования. Его общий вид приведен на рисунке 11. Данный узел укладывает коробки на поддон.



Рис. 11 – Установка по паллетизации творогов

Далее рассмотрим задачу транспортировки готового продукта на склад с помощью мобильного робота. Вид робота приведен на рисунке 12.



Рис. 12 – Мобильный транспортный узел

Мобильный робот оперирует понятием миссии, которая состоит из точки забора продукта, маршрута доставки и точки выгрузки. После запуска производственного процесса робот занимает позицию погрузки или ожидает внешнего сигнала о том, что паллета заполнена. После этого робот приступает к процессу загрузки паллеты

на борт. Далее, основываясь на доступной карте помещения, динамической обстановке в цеху с соблюдением норм безопасности, робот передвигается с паллетой до приемного порта, где при наличии свободного места осуществляет разгрузку.

Конструирование миссий для робота выполняется системой комплексной автоматизации. Для этого решаются следующие подзадачи:

- создание расписания для вывоза продукта с нескольких линий с учетом производительности их и робота;
- выбор времени зарядки мобильного робота в моменты простоя;
- определение уровня безопасности по уровню текущей динамической ситуации в цехе.

VIII. ЗАКЛЮЧЕНИЕ

В работе рассмотрен онтологический подход к проектированию предприятий рецептурного производства, предполагающий переход от рассмотрения производственных процессов в рамках одного предприятия к рассмотрению процессов, охватывающих, в соответствии с концепцией Industry 4.0, и смежные предприятия – поставщиков сырья (молочные фермы) и оптовых потребителей продукции (магазины, торговые сети).

Рассмотрены примеры реализации указанного подхода на предприятии ОАО «Савушкин продукт», в частности, рассматривается описание моделей предприятия на основе онтологического подхода, реализация подсистемы информационного обслуживания сотрудников в рамках SCADA-системы, подсистема управления логистическими процессами, а также подсистемы, ориентированной на работу с роботизированными установками.

1. Голенков, В. В. Проектирование предприятий рецептурного производства на основе онтологий / В. В. Таберко, Д. С. Иванюк, В. В. Голенков, К. В. Русецкий, Д. В. Шункевич, И. Т. Давыденко, В. В. Захаров, В. П. Ивашенко, Д. Н. Корончик // *Онтология проектирования*. – 2017. – Т. 7, №2(24). – С. 123–144. – DOI: 10.18287/2223-9537-2017-7-2-123-144.
2. Интеграция искусственных нейронных сетей с базами знаний / В. В. Голенков и другие // *Онтология проектирования*. – 2018. – Т. 8, №3(29). – С. 366–386.
3. Taberko, V. Design Principles of Integrated Information Services for Batch Manufacturing Enterprise Employees / V. Taberko [et al.] // *Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2019): материалы международной научно-технической конференции, Минск, 21–23 февраля 2019 г.* / Белорусский государственный университет информатики и радиоэлектроники; редкол.: В. В. Голенков (гл. ред.) [и др.]. – Минск, 2019. – С. 215–224.
4. Recommendations for Implementing the Strategic Initiative Industrie 4.0: Securing the Future of German Manufacturing Industry / H. Kagermann, J. Hellwig, A. Hellinger, W. Wahlster // *Final Report of the Industrie 4.0 Working Group*. – 2013.
5. Hermann, M, Pentek, T, Otto, B. Design principles for industrie 4.0 scenarios. In *System Sciences (HICSS): 49th Hawaii International Conference on 2016 Jan 5*. – 2016. – Pp. 3928–3937.
6. *Applied Cyber-Physical Systems* / Ed. by S. C. Suh, U. J. Tanik, J. N. Carbone, A. E. Eroglu. – Heidelberg: Springer-Verlag. – 2014.
7. Тарасов, В. Б. Стратегический инжиниринг предприятий будущего: массовое сотрудничество, интернет вещей, инициатива «индустрия 4.0», что дальше? // *Инжиниринг предприятий и управление знаниями (ИП&УЗ-2016): сб. науч. тр. XIX науч.-практ. конф. Москва, 26–27 апреля 2016 г.* / под науч. ред. Ю. Ф. Тельнова. М.: ФГБОУ ВО «РЭУ им. Г. В. Плеханова». – 2016. – С. 57–68.
8. [Electronic resource] / Garbrecht, S. The three rules of Industrial Operations Management and Industrial IoT Applications. – 2017. – Mode of access: <https://www.linkedin.com/pulse/three-rules-industrial-operations-management-iot-steven-garbrecht/>. – Date of access: 14.09.2021.
9. Тарасов, В. Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика / В. Б. Тарасов. – М.: Эдиториал УРСС, 2002. – 348 с.
10. [Electronic resource] / ISA88, Batch Control. – 2016, Nov. – Mode of access: <https://www.isa.org/isa88>. – Date of access: 03.09.2021.
11. Гаврилова, Т. Л. и др. Инженерия знаний. Модели и методы: Учебник / Т. Л. Гаврилова, Д. В. Кудрявцев, Д. И. Муромцев. – СПб.: Издательство «Лань», 2016. – 348 с.
12. Шведин, Б. Я. Онтология предприятия: экспириентологический подход: Технология построения онтологической модели предприятия / Б. Я. Шведин. – М.: Ленанд, 2010. – 240 с.
13. Поспелов, Д. А. Ситуационное управление. Теория и практика / Д. А. Поспелов. – М.: Наука, 1986. – 284 с.
14. Robson, M., Ullah, P. A practical guide to business process re-engineering, Gower Publishing, Ltd. – 1996. – 171 p.
15. Instrumentation, Systems, and Automation Society, ANSI/ISA-88.02-2001 Batch Control Part 2: Data Structures and Guidelines for Languages, ISA, Research Triangle Park, USA. – 2001.
16. Голенков, В. В. Открытая технология онтологического проектирования, производства и эксплуатации семантически совместимых гибридных интеллектуальных компьютерных систем / В. В. Голенков, Н. А. Гулякина, Д. В. Шункевич. – Минск: Бестпринт, 2021. – 690 с. – ISBN 978-985-7267-13-2.

Секция «Системы управления»

Section «Control systems»

МОДЕЛЬ ПРОГНОЗИРОВАНИЯ ВРЕМЕНИ ТЕСТИРОВАНИЯ ПРИКЛАДНЫХ КОМПЬЮТЕРНЫХ ПРОГРАММ ДЛЯ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ

Боровиков С. М., Казючиц В. О., Дик С. С., Юрениа К. В.
Кафедра проектирования информационно-компьютерных систем,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: {bsm, vladkaz}@bsuir.by, kytech@yandex.by

Прикладные компьютерные программы, используемые в автоматизированных системах управления, содержат сотни тысяч-миллионы строк программного кода и после написания этих программ в них всегда имеются скрытые ошибки, определяющие уровень надёжности программ. Уровень надёжности компьютерной программы перед началом её тестирования обычно не отвечает требованиям практики. Задача этапа тестирования состоит в том, чтобы выявить и устранить наиболее критичные ошибки и свести долю скрытых ошибок к минимуму. Возникает вопрос, сколько времени необходимо выполнять тестирование. На основе статистических данных о тестировании и надёжности компьютерных программ разного функционального назначения, в том числе используемых в автоматизированных системах управления, получена модель, позволяющая определять прогнозируемое время тестирования, обеспечивающее для разрабатываемой программы заданный уровень её эксплуатационной надёжности.

Работа проводилась в рамках договора № Ф20МВ-021 на выполнение НИР «Статистические модели надёжности прикладных программных средств и их использование для оценки ожидаемой безотказности компьютерных программ на ранних этапах их разработки» в соответствии с решением научного совета БРФФИ по результатам конкурса «БРФФИ-Минобразование М-2020 (протокол № 1 от 22.04.2020).

ВВЕДЕНИЕ

Автоматизированные системы управления (АСУ) относятся к сложным информационно-компьютерным системам, в которых вклад программного обеспечения в ненадёжность систем может составлять 40 и более процентов [1]. Качество работы АСУ зависит от надёжности используемых прикладных компьютерных программ. Для определения целесообразности разработки и оценки эффективности работы проектируемой АСУ в заданных условиях необходимо на ранних этапах её проектирования оценить ожидаемую эксплуатационную надёжность планируемых к разработке в составе АСУ прикладных компьютерных программ. Под эксплуатационной надёжностью компьютерной программы будем понимать тот уровень надёжности, который она покажет на начальном этапе эксплуатации, то есть после выполнения этапа тестирования и отладки программы.

I. АКТУАЛЬНОСТЬ

Прикладные компьютерные программы, используемые в АСУ, содержат сотни тысяч-миллионы строк программного кода и поэтому после написания программ в них всегда имеются скрытые ошибки. Наличие этих ошибок определяют уровень надёжности компьютерных программ. После написания кода компьютерной программы и устранения нарушений правил языка программирования программа характеризуется начальным уровнем надёжности, который

как правило не отвечает требованиям практики. Поэтому далее выполняют этап тестирования и отладки программы. Задача этого этапа состоит в том, чтобы выявить и устранить наиболее критичные ошибки, свести к минимуму долю скрытых ошибок, оставшихся в компьютерной программе, и в итоге обеспечить приемлемый уровень эксплуатационной надёжности компьютерной программы. Разработчики АСУ хотели бы знать ожидаемый уровень эксплуатационной надёжности разрабатываемой прикладной компьютерной программы ещё до написания кода программы и выполнения её тестирования. Разработчиков АСУ интересуют также финансовые затраты и/или затраты времени, которые потребуются для достижения заданного уровня надёжности прикладной компьютерной программы.

II. ИСХОДНЫЕ ПОЛОЖЕНИЯ И ПРЕДПОСЫЛКИ

Оценка надёжности прикладной компьютерной программы до написания её программного кода была рассмотрена в [2]. Подход основан на статистических моделях ожидаемого числа возможных ошибок в компьютерной программе (некоторые специалисты используют термин «дефекты проектирования программы»), и использовании некоторых параметров будущей компьютерной программы, а также данных о влиянии на её надёжность различных факторов и предполагаемое в дальнейшем тестирова-

ние компьютерной программы:

$$\lambda_{exp} = \frac{60}{Q} K_s \frac{R}{B} F_0 L * 10^{-6} = \frac{60}{Q} K_s V N_0 * 10^{-6}, \quad (1)$$

где λ_{exp} - ожидаемая эксплуатационная интенсивность отказов ($ч^{-1}$) прикладной компьютерной программы; Q - коэффициент эффективности тестирования компьютерной программы; K_s - суммарный коэффициент увеличения интенсивности отказов, обусловленный совместным действием изменчивости входных данных и рабочей нагрузки на компьютерную программу; R - пиковое быстродействие процессора, указываемое производителем в технической документации, размерность: операций в секунду; B - количество команд (операторов) в компьютерной программе; F_0 - начальная плотность ошибок в программе; L - объём компьютерной программы в исполняемых строках программного кода; V - скорость выполнения команд компьютерной программы (скорость для пикового быстродействия процессора R), определяемая отношением $V = R/B$; N_0 - число ошибок в компьютерной программе до выполнения тестирования. Параметр F_0 в выражении (1) - это плотность ошибок компьютерной программы после написания её кода и устранения нарушений правил языка программирования, представляет собой среднее число ошибок, приходящихся на строку кода: $F_0 = N_0/L$. Модель (1) учитывает отрасль применения компьютерной программы, быстродействие процессора компьютера, изменчивость входных данных, а также рабочую нагрузку, которую будет воспринимать компьютерная программа со стороны эксплуатационной среды (ввод-вывод данных и нахождение этих операций в очереди, наличие состояний ожиданий, загрузка-выгрузка программы и/или её модулей из памяти и т.д.). Рассматривается получение параметров модели. Из работы [2] следует, что для достижения заданного уровня эксплуатационной надёжности компьютерной программы необходимо обеспечить нужное значение коэффициента эффективности тестирования Q , определяемого отношением [2].

$$Q = \frac{N_0}{N_t}, \quad (2)$$

где N_t - число оставшихся ошибок в прикладной компьютерной программе после завершения её тестирования.

III. МОДЕЛЬ ПРОГНОЗИРОВАНИЯ ВРЕМЕНИ ТЕСТИРОВАНИЯ КОМПЬЮТЕРНОЙ ПРОГРАММЫ

Для получения модели прогнозирования, используемой для определения времени, требуемого для проведения тестирования планируемой к разработке прикладной компьютерной программы, необходимо величину $N_{п.тест}$ в отношении (2) выразить через время тестирования t и далее решить полученное уравнение относительно t . В докладе с использованием модели (1), отношения (2) и положений, приводимых в [3], модель для прогнозирования времени тестирования компьютерной программы получена в виде

$$t = \frac{LE_L \ln(Q)}{60K_s R}, \quad (3)$$

где E_L - коэффициент расширения кода, показывающий увеличение числа команд компьютерной программы относительно числа строк программного кода L . Значение E_L зависит от используемого языка программирования, например для *C* $E_L = 2,5$; для *Fortran*, *Cobal* $E_L = 3,0$; для *Ada* $E_L = 4,5$; для *C++* $E_L = 6,0$. Коэффициент K_s , рассчитанный по экспериментальным данным [4] для компьютерных программ, используемых для АСУ, принял значение $K_s = 19,2$. В модели (3) в качестве t рассматривается процессорное время выполнения компьютерной программы. В дальнейшем при необходимости время выполнения компьютерной программы t может быть преобразовано в календарное время.

IV. СПИСОК ЛИТЕРАТУРЫ

1. Чуканов, В. О. Методы обеспечения аппаратно-программной надежности вычислительных систем / В. О. Чуканов, В. В. Гуров, Е. В. Прокопьева [Электронный ресурс]. - 2014. - Режим доступа : <http://www.mcst.ru/metody-obespecheniya-apparatnoprogrammnoj-nadezhnosti-vychislitelnykh-sistem>. - Дата доступа: 4 апреля 2021.
2. Borovikov S. M., Kaziuchyts V. O., Khoroshko V. V., Dick S. S., Klinov K. I. Assessment of expected reliability of applied software for computer-based information systems. *Informatics*, 2021, vol. 18, no. 1, pp. 84–95 (in Russian). <https://doi.org/10.37661/1816-0301-2021-18-1-84-95>
3. Шубинский, И. Б. Функциональная надёжность информационных систем. Методы анализа / И. Б. Шубинский. - М. : «Журнал Надежность», 2012. - 296 с.
4. McCall, J. A. *Software Reliability, Measurement, and Testing Guidebook for Software Reliability Measurement and Testing* [Electronic resource] / J. A. McCall [et al.]. - 1992. - Mode of access: <https://apps.dtic.mil/dtic/tr/fulltext/u2/a256164.pdf>. - Date of access: 20.10.2021.

НАСТРОЙКА ПИД-РЕГУЛЯТОРА ДЛЯ СИСТЕМЫ С НЕЛИНЕЙНОЙ ДИНАМИКОЙ

Гринюк Д. А., Олиферович Н. М., Сухорукова И. Г., Оробей И. О.
 Кафедра автоматизации производственных процессов и электротехники,
 Белорусский государственный технологический университет
 Минск, Республика Беларусь
 E-mail: hryniukda@gmail.com

Рассмотрено решение по моделированию объектов с нелинейной динамикой. Целью исследования являются системы регулирования с параметрической зависимостью динамических характеристик. В качестве примера выбран объект управления второго порядка с линейной зависимостью постоянной времени от выходного параметра. Идентификация объекта с целью получения линейной модели при различных направлениях отклонения приводит к различным результатам. Для настройки ПИД-регулятора использован интегральный критерий качества.

ВВЕДЕНИЕ

Нелинейные свойства проявляются у большинства объектов управления технологическими процессами. Обычно проблему решают с помощью линеаризации, выбор которой зависит от задач и свойств объекта. В некоторых случаях такой подход не позволяет качественно решить задачу настройки системы. Вариантом решения проблемы может быть перенастраивание системы применяя нелинейное управление. Например, нечеткие, нейронные, гибридные подходы, варианты математического моделирования с формированием целевой функции и поиском ее экстремума.

Ряд технологических объектов имеет динамику, зависящую от параметра, который непосредственно регулируется. К таким объектам можно отнести, например, стабилизацию уровня L . Постоянная времени обычно определяется как

$$T = \frac{LS}{F}, \quad (1)$$

где S – площадь сосуда, танка; F – расход.

Вариант, при котором физический параметр является одновременно управляющим воздействием и используется для определения постоянной времени, характерен и для тепловых процессов.

$$T = \frac{WC}{Q}(\Theta - \Theta_w), \quad (2)$$

где W – масса теплообменной стенки; C – теплоемкость; Q – количество тепла, которое участвует в теплообмене; Θ – температура теплоносителя; Θ_w – температура стенки.

I. МОДЕЛИРОВАНИЕ

В случае, если при определении постоянных времени нельзя использовать значение у стационарной точки (например, постоянные времени, рассчитанные по формулам (1)–(2), изменяются в широком диапазоне), может быть предложена структура моделирования как на рис.1., где $x(s)$ – вход; $y(s)$ – выход; k_1, k_2 – коэффициенты

настройки диапазона изменения постоянной времени. Коэффициент k_2 , в первую очередь, определяет начальное значение постоянной времени; k_1 – диапазон изменения. Предложенная структура демонстрирует линейную зависимость постоянной времени от выходного параметра. При необходимости можно использовать и нелинейные функции при моделировании динамики.

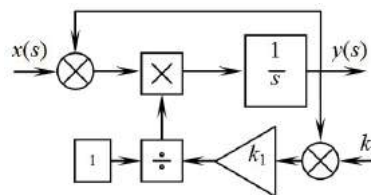


Рис. 1 – Структура для моделирования динамики объекта, которая зависит от выходного параметра

На базе структуры рис.1 было произведено моделирование объекта управления второго порядка с различным диапазоном изменения параметров k_1 и k_2 . Результаты моделирования аппроксимировались звеном второго порядка с постоянными коэффициентами. Отличие между моделированием линейной (аппроксимированной) и нелинейной (исходной) системы можно видеть на рис.2. Диапазон изменений постоянной времени исходной структуры $TV_1 = 60-200$ и $TV_2 = 30-50$.

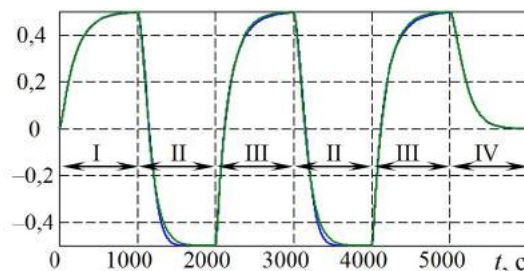


Рис. 2 – Переходной процесс объекта с переменной динамикой и результат его аппроксимации

Идентификация производилась путем аппроксимации тренда переходным процессом звена второго порядка с постоянными времени TC_1

и TC_2 и минимизацией среднеквадратичного отклонения. Коэффициенты получались отдельно, для каждого из тактов I–IV (рис. 2). Для выбранных вариантов оценивалось интегральное значение среднеквадратичного отклонения (ISD). Чтобы облегчить анализ, для каждого из вариантов определялось относительное отклонение $RISD$ по отношению к минимальному значению ISD . Были рассмотрены 6 исходных вариантов значений параметров модели рис. 1: [$TV_1 = 80-160$ $TV_2 = 30-50$]; [$TV_1 = 60-200$ $TV_2 = 30-50$]; [$TV_1 = 100-120$ $TV_2 = 30-50$]; [$TV_1 = 80-160$ $TV_2 = 35-45$]; [$TV_1 = 80-160$ $TV_2 = 15-65$]; [$TV_1 = 100-120$ $TV_2 = 15-65$].

Результаты аппроксимировались хуже всего при несовпадении направления воздействия на объект и градиента зависимости динамики (II). В этом случае, для улучшения аппроксимации необходимо увеличивать порядок передаточной функции. Интегральное значение среднеквадратичного отклонения отличалось от диапазона и направления отклонения (табл. 1). Чем меньше отклонение, тем лучше качество аппроксимации. Полученные результаты хорошо коррелируются с результатами прикладных экспериментов с тепловыми объектами [1, 2].

Таблица 1 – Минимальное значение среднеквадратичного отклонения ISD

Эксперимент	Значение ISD
$TV_1 = 80-160$ и $TV_2 = 30-50$	0,000220
$TV_1 = 60-200$ и $TV_2 = 30-50$	0,00447
$TV_1 = 100-120$ и $TV_2 = 30-50$	0,000269
$TV_1 = 80-160$ и $TV_2 = 35-45$	0,000219
$TV_1 = 80-160$ и $TV_2 = 15-65$	0,000326
$TV_1 = 100-120$ и $TV_2 = 15-65$	0,000272

II. ПОИСК НАСТРОЕК ПИД-РЕГУЛЯТОРА

Настройку систем управления для объектов с переменной динамикой можно производить из разных позиций. Самым простым является нахождение таких настроек, при которых система будет устойчива и обеспечит приемлемое качество управления в диапазоне изменяющейся динамики. Исходя из полученных результатов, видим, что подход хорош при небольшом диапазоне отклонений выходной величины. В противоположном случае встает более сложный вопрос, так как динамика вверх и вниз может иметь существенные отличия.

В современных условиях задача поиска настроек регулятора часто решается использованием адаптивных методов, нечетких алгоритмов, нейронных систем и т.д. Однако в случае детерминированного диапазона изменения динамики, использование фиксированных настроек или варианта табличных настроек будет предпочтительным.

В качестве примера был выбран вариант, при котором $TV_1 = 80-160$ и $TV_2 = 30-50$. Настройка проводилась для нелинейной модели в

том же диапазоне изменения выходной величины, что и при идентификации по интегральному критерию [3–4]:

$$\int_0^{tf} t^2 |e(t)| dt \rightarrow \min; \quad (3)$$

где $e(t)$ – невязка между сигналом задания и выходной переменной; tf – время от изменения сигнала задания до окончания переходного процесса.

Поиск настроек осуществлялся для структуры на рис.2. Передаточная функция с постоянной времени 10 выполняла роль датчика. Настройки, которые были найдены путем минимизации критерия (3) на базе нелинейной модели и результатов идентификации показали различные результаты (Таблица. 2). Использование нелинейной модели при поиске настроек позволяет получить более универсальные настройки.

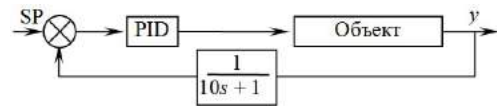


Рис. 3 – Структура системы

Таблица 2 – Результаты настройки ПИД-регулятора

Объект	k_p	T_I	T_D
Идентификация по II	6,75	23,90	106,5
Идентификация по III	14,41	11,26	587,9
Нелинейная структура	2,67	58,05	60,07

III. ЗАКЛЮЧЕНИЕ

Современные системы моделирования предоставляют широкие возможности по настройке систем управления наилучшим образом с учетом реальных свойств объектов управления. Предложенная структура моделирования может быть использована для широко класса объектов с нелинейными свойствами в диапазоне управления.

1. Marozava, M. Experimental study of the variation dynamic's for air heat exchanger /M. Marozava, D. Hryniuk // Mokslas – Lietuvos ateitis / Science – Future of Lithuania. 2017. Vol. 9, no. 3. P. 297–301.
2. Гринюк Д. А. Оценка динамики изменения температуры по длине металлического стержня / Д. А. Гринюк [и др.] // Автоматизация и энергосбережение машиностроительного и металлургического производств, технология и надежность машин, приборов и оборудования: материалы XIII Междунар. на-уч.-техн. конф. – Вологда: ВоГУ, 2018. С.85–88.
3. Hryniuk D., Non-linear PID controller and methods of its setting / D. Hryniuk, I. Suhorukova, I. Orobei // Open Conference Electrical, Electronic and Information Sciences (eStream). Vilnius, 2017. P. 1–4. DOI: 10.1109/eStream.2017.7950327.
4. Сухорукова, И. Г. Применение в ПИД-законе регулирования нелинейных функций преобразования ошибки /И. Г. Сухорукова, Д. А. Гринюк, И. О. Ороей // Труды БГТУ. 2013. № 6: Физ.-мат. науки и информатика. С. 95–98.

ПОСТАНОВКА ЗАДАЧ ГРУППОВОГО УПРАВЛЕНИЯ ПРОМЫШЛЕННЫМИ МОБИЛЬНЫМИ РОБОТАМИ

Довнар А. Д.
Кафедра систем управления,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: dovnar@bsuir.by

Разработка систем управления группами роботов не является тривиальной. Она состоит из большого количества отдельных направлений и подзадач. У начинающих разработчиков и исследователей может возникнуть большое количество проблем в понимании глобальной картины. Это может, привести к неправильному выбору конкретного направления деятельности, плохой оценке трудоемкости проектов на ранних этапах или вовсе к потере интереса к области. Эта работа написана для того, чтобы разъяснить задачи, возникающие в процесс разработки систем управления группами промышленных мобильных роботов.

ВВЕДЕНИЕ

Разработка роботов находится на стыке большого количества направлений деятельности. К 4 основным можно отнести:

- электронику;
- механику;
- программирование;
- системы управления.

По определению ISO 8373[1], подвижное устройство относится к **робототехнике** только если имеет определенную степень автономности. Под автономностью подразумевается возможность устройства выполнять поставленные задачи, основываясь на текущем состоянии и показаниях датчиков без вмешательства человека. Разработка неавтономных подвижных устройств называется **мехатроникой**. Обычно задачи мехатроники проще задач робототехники и делают упор на механику и электронику. Упор в робототехнике делается преимущественно на разработку алгоритмов управления и программного обеспечения, реализующего эти алгоритмы. Таким образом задачу управления группой мобильных роботов можно сформулировать как, задачу управления группой автономных подвижных объектов.

Разобьем задачи управления группами мобильных роботов на две категории:

- управление отдельными роботами;
- управление группой роботов.

I. СПЕЦИФИКА ПРОМЫШЛЕННЫХ ГРУЗОВЫХ РОБОТОВ

Задачи группового управления роботами во многом идентичны для различных групп роботов, например, летающих, плавающих, мобильных. Однако каждая отдельная группа обладает своими особенностями. Эти особенности могут добавлять дополнительные ограничения или требования из-за чего их стоит рассматривать отдельно.

В работе описаны задачи группового управления промышленными мобильными грузовыми наземными роботами. Роботы компаний OTTO и MIR — хорошие примеры подобных устройств.

Рабочей средой для таких роботов являются складские или производственные помещения. Основные особенности рабочей среды:

- ровные полы;
- относительно хорошая предсказуемость: мало движущихся объектов, карта помещения заранее известна и не меняется;
- контролируемый микроклимат, отсутствие погодных условий;
- отсутствие изменений высоты. Могут быть только изменения этажей, но перемещение между этажами происходит только в лифтах;
- возможность размещать ориентиры на заведомо известных позициях в пространстве.

Данные особенности делают промышленную среду более простой чем, например, открытое пространство на улице.

II. УПРАВЛЕНИЕ ОТДЕЛЬНЫМИ РОБОТАМИ

Возможность группового управления должна учитываться еще на этапе разработки механики и электроники. Очень важно учитывать параметры рабочей среды, в которой роботы будут работать и особенности их использования.

Обычно, обеспечение группового управления промышленными роботами требует наличия следующих особенностей конструкции:

- мощную систему связи для обмена данными с другими роботами и/или центральной системой;
- сенсорную избыточность (больше сенсоров, тем теоретически нужно для эффективной навигации) для увеличения надежности при взаимодействии с другими роботами и людьми;

- мощную вычислительную систему для обработки данных, получаемых от сенсоров и других роботов.

Кроме этого появляются дополнительные требования к инфраструктуре, например, необходимость наличия производительного сервера для обработки данных, получаемых от группы роботов.

После разработки физического робота актуальной проблемой становится разработка системы управления. Можно выделить несколько основных задач.

Разработка локальной системы управления. То есть как заставить робота двигаться и выполнять требуемые функции. К данной задаче относятся:

- разработка драйверов датчиков;
- разработка систем управления отдельными объектами, например двигателями;
- разработка системы управления движением робота в целом.

Разработка системы навигации. То есть как роботу понять, где он находится. К данной задаче относятся:

- комбинирование показаний разных навигационных датчиков между собой;
- определение текущего положения робота в пространстве, например относительно известной глобальной карты или точки включения робота;
- построение карты местности.

Разработка локального планировщика маршрута. То есть как роботу проехать из точки А в точку Б. Данная задача подразумевает определение желаемой траектории движения. Локальный планировщик получает команды управления извне, например, от глобального планировщика маршрута, пользователя или внешнего мира. Локальный планировщик должен проложить траекторию движения максимально близкую к заданию, но при этом избежать столкновений и других опасных ситуаций.

III. УПРАВЛЕНИЕ ГРУППОЙ РОБОТОВ

После того, как робот самостоятельно передвигается в пространстве, можно начинать работать над групповым управлением несколькими роботами.

Можно выделить несколько основных задач управления группой роботов [2].

Разработка системы глобальной навигации. То есть как роботам понять где они на-

ходятся относительно мира и друг друга, как поделиться этими данными с другими роботами.

Разработка глобального планировщика маршрута. То есть как понять, куда и как должен двигаться каждый отдельный робот в группе.

Разработка алгоритмов взаимодействия. То есть как роботам выполнять совместные задачи. В данном направлении есть огромное множество задач, например движение за лидером, движение в формации, сканирование местности и так далее. Конкретные алгоритмы разрабатываются в зависимости от целей проекта.

Для оценки алгоритмов взаимодействия роботов используются следующие критерии[3]:

- локальность – используется для управления роботом только локальная информация (хорошо) или также глобальная (плохо);
- масштабируемость – работает ли алгоритм одинаково хорошо с несколькими роботами и несколькими десятками роботов;
- безопасность – обеспечивает ли алгоритм возможность избежать столкновений между роботами и роботов с препятствиями;
- децентрализованность – используются ли для алгоритма вычисления на центральном сервере (плохо) или все вычисления проводятся локально (хорошо).

IV. ЗАКЛЮЧЕНИЕ

Все задачи описанные в этой работе: разработка конструкции роботов, локальной системы управления и групповой системы управления выполняются каждый раз когда разрабатывается новый промышленный грузовой робот. Любая из этих задач может стать направлением для углубленного изучения.

V. СПИСОК ЛИТЕРАТУРЫ

1. ISO 8373:2012(en), Robots and robotic devices – Vocabulary [Электронный ресурс]. Режим доступа: <https://wcvw.iso.org/obp/ui/#iso:std:iso:8373:ed-2:vl:en> Дата доступа: 18.10.2021.
2. Tkafestas S. G. Mobile Robot Control and Navigation: A Global Overview // J Intell Robot Syst. – 2018. – Т. 91. – № 1. – С. 35-58.
3. Pickem D. и др. The Robotarium: A remotely accessible swarm robotics research testbed // 2017 IEEE International Conference on Robotics and Automation (ICRA). Singapore, Singapore: IEEE, – 2017. С. 1699-1706.

КИНЕМАТИКА И ДИНАМИКА НОЖНИЧНОГО ПОДЪЕМНИКА

Довнар А. Д., Рогач А. Ю.
Кафедра систем управления,

Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: dovнар@bsuir.by. aleksandra.rogacli256@gmail.com

Разработка, ножничных подъемников является актуальной и сложной задачей. Одной из проблем управления, ножничным подъемником является, нелинейность кинематики и динамики механизма. В данной работе приведен вывод кинематических и динамических законов движения, на основании которых возможно разработать систему управления.

ВВЕДЕНИЕ

Ножничные подъемники активно используются при строительных работах, погрузках и разгрузках на производстве, автосервисе, а также в робототехнике. Их нелинейное движение может приводить к дискомфорту пассажиров, а также усложняет перемещение на небольшие дистанции. Для обеспечения линейности движения необходимо разработать систему управления, для чего нужно знать кинематику и динамику процесса.

I. ОПИСАНИЕ МЕХАНИЗМА

Ножничный подъемник — это тип подъемного механизма, состоящий из двух наборов крестообразных секций [1] (рисунок 1). Два параллельных набора необходимо для обеспечения устойчивости. Каждая секция состоит из двух балок, соединенных между собой в центре. В зависимости от требуемой высоты может использоваться разное количество секций. Данная работа описывает тип подъемника, состоящий из одной секции. Секция сверху и снизу имеет по два крепления. Одно из креплений шарнирно закреплено: может вращаться вокруг оси крепления, но не может смещаться в других направлениях. Второе крепление может свободно перемещаться вдоль секции. В данной работе описывается подъемник, приводимый в движение благодаря линейному механизму, закрепленному внизу секции.

Два самых распространенных типов приводов: гидравлический и винтовая передача, гидравлические приводы применяются при высокой грузоподъемности, а винтовая передача для более точного управления и меньших габаритов механизма. В качестве линейного механизма в данной работе используется винтовая передача, приводимая в движение электрическим двигателем. Гайка передачи присоединена к подвижному концу балки. Движение гайки приводит в движение присоединенную балку, что в свою очередь изменяет высоту механизма.

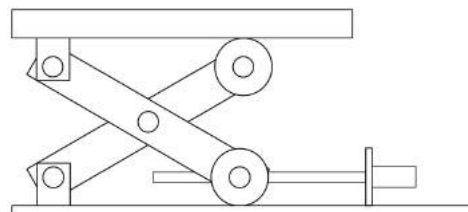


Рис. 1 – Схема ножничного подъемника

II. КИНЕМАТИКА ПОДЪЕМНИКА

Для описания механизма введем обозначения, показанные на рисунке 2.

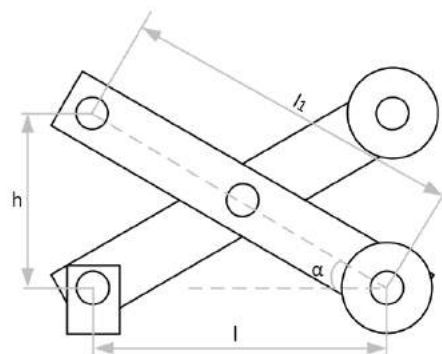


Рис. 2 – Введенные обозначения

Где l — расстояние между креплениями (управляемая величина), l_1 — длина балки подъемника (постоянная величина), α — угол между основанием подъемника и балкой (переменная величина), h — текущая высота одной секции подъемника (переменная величина).

Обе балки описывают окружности вокруг своих точек закрепления на основании. Соответственно, максимальное теоретически возможное значение $l = l_1$, а минимальное 0. Точно по этой же причине максимальная высота секции $h = l_1$, а минимальная 0. Следовательно, угол α находится в интервале $[0^\circ; 90^\circ]$.

Для получения уравнений кинематики необходимо проанализировать треугольник, в котором основание и высота подъемника являются катетами, а одна из балок — гипотенузой (рисунок 2). Из треугольника можно получить следующие уравнения:

$$\sin \alpha = \frac{\sqrt{c_1^2 - l_2}}{l_1} \quad (1)$$

$$\cos \alpha = \frac{l}{l_1} \quad (2)$$

$$h(l) = n \cdot \sqrt{l_1^2 - l^2} = n \cdot l_1 \cdot \sin \alpha \quad (3)$$

Скорость подъема можно найти, взяв производную (3) по l или α . Формулы по l и α взаимозаменяемы. По α удобнее проводить анализ, а l используется для управления.

$$h'(l) = -n \cdot \frac{l}{\sqrt{l_1^2 - l^2}} \cdot l' \quad (4)$$

$$h'(\alpha) = n \cdot l_1 \cdot \cos \alpha \cdot \alpha' \quad (5)$$

В формулах (4) и (5) видно, что скорость подъема зависит не только от скорости изменения управляющего параметра (угла или расстояния), но и от текущего значения параметра. Это поведение нелинейно.

III. ДИНАМИКА ПОДЪЕМНИКА

В формулах ниже выведена подъемная сила механизма, направленная вертикально вверх. Учитывая, что система полностью симметричная, можно предположить, что любой силе будет соответствовать сила такая же по величине, но обратная по направлению. Если ввести это предположение, сделать проекцию создаваемой винтом силы на балку и проекцию полученной силы на груз, а также учесть формулы (1) и (2), можно получить следующие выражения.

$$F(\alpha) = \frac{F_{acrew}}{2} \cdot \sin(2 \cdot \alpha) \quad (6)$$

$$F(l) = F_{acrew} \cdot \frac{l \cdot \sqrt{l_1^2 - l^2}}{l_1^2} \quad (7)$$

где F_{acrew} – сила, создаваемая винтом.

Из формул (6) и (7) видно, что подъемная сила равна 0, когда $l = l_1$ (то есть $h = 0$) и $l = 0$ (то есть $h = l_1$). Для движения в этих точках необходима бесконечная сила на винте.

Преобразование силы в подъемнике нелинейное. Это означает, что добиться линейного подъема будет сложно, так как надо будет добиться линейного ускорения.

Для получения уравнения динамики механизма воспользуемся вторым законом Ньютона. Для этого введем несколько упрощений:

- трение отсутствует;
- масса балок очень мала по сравнению с массой груза.

Соответственно, получаем следующее выражение:

$$m \cdot \alpha = F - m \cdot g,$$

где m – масса груза на подъемнике, α – ускорение массы, находящейся на подъемнике, F –

сила, действующая на массу от подъемного механизма.

Если вместо F подставить выражение (7), получим:

$$m \cdot \alpha = F_{acrew} \cdot \frac{l \cdot \sqrt{l_1^2 - l^2}}{l_1^2} - m \cdot g$$

Соответственно, чтобы добиться линейности необходимо управлять силой по следующему закону [2]:

$$F_{acrew} = \frac{K \cdot l_1^2}{l \cdot \sqrt{l_1^2 - l^2}} \quad (8)$$

где K необходимая для конкретного случая сила (для достижения определенной скорости и/или ускорения).

На рисунке 3 приведен график функции (8):

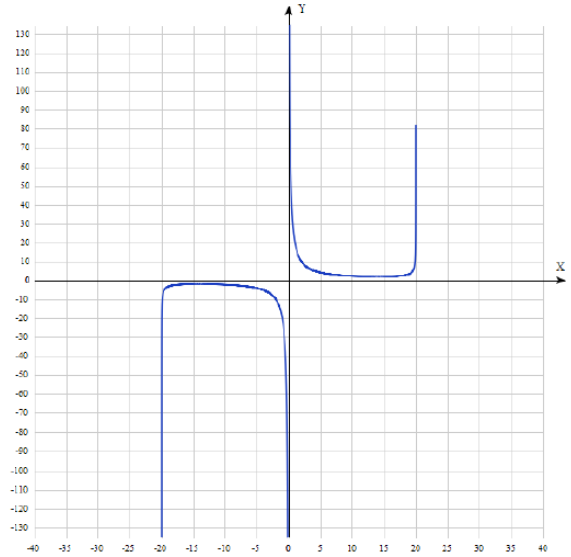


Рис. 3 – График функции F_{acrew}

На графике видно, что на двух краях диапазона положений есть места, где для движения необходима бесконечная сила. Также видно, что есть относительно линейный участок, на котором можно работать с «адекватным» КПД, то есть не затрачивая слишком много силы.

IV. ЗАКЛЮЧЕНИЕ

В данной работе описаны вывод законов кинематики и динамики, которые можно использовать для разработки системы управления ножничным подъемником.

V. СПИСОК ЛИТЕРАТУРЫ

1. Saxona A. Deriving a Generalized, Actuator Position-Independent Expression for the Force Output of a Scissor Lift C. 18.
2. Zhang W. и др. A Study on the Static Stability of Scissor Lift // TOME.I. 2015. – Т. 9. – No 1. – С. 954-960.

МОБИЛЬНОЕ ПРОГРАММНОЕ СРЕДСТВО ДЛЯ АВТОМАТИЗАЦИИ ВЗВЕШИВАНИЙ ЖЕЛЕЗНОДОРОЖНЫХ ВАГОНОВ

Дорох А. А., Деменковец Д. В

Кафедра программного обеспечения информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: alexeydoroh@gmail.com, Demenkovets@bsuir.by

В докладе рассматривается состав программно-аппаратного комплекса, программное средство автоматизации процесса динамического взвешивания железнодорожных вагонов.

ВВЕДЕНИЕ

Учет груза, перевозимых по железной дороге, имеет большое значение для правильной эксплуатации подвижного состава. Данный процесс играет значимую роль для развития предприятий.

Таким образом задача организации процессов учета грузов, перевозимых железнодорожным транспортом, является крайне востребованной и актуальной на сегодняшний день. Автоматизация процесса взвешивания железнодорожных вагонов, как в статическом режиме, так и в движении, является одним из основных направлений развития логистики [1]. Существенным преимуществом для автоматизации данного процесса будет мобильное приложение, позволяющее ускорить и облегчить процесс взвешивания оператору подвижных составов в статическом режиме с обработкой полученных данных и автоматическим формированием отчетов на основе этих данных.

I. ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС

В данное время широко применяются статодинамические весы, позволяющие производить взвешивание как в статическом режиме с остановкой каждого вагона на весовой платформе, так и в движении на скорости до 15 км/ч. Тем не менее, в данном режиме требуется специальная настройка весового оборудования для минимизации погрешностей полученных данных о взвешиваниях.

При установке вагона на измерительную платформу, параметры взвешивания передаются от весового контроллера на персональный компьютер оператора по интерфейсу Ethernet с помощью протоколов TCP/IP. Уведомление машиниста локомотива о статусе процесса взвешивания производится посредством отображения данных на информационном табло [2]. Но из-за того, что ПК оператора находится в помещении, он ограничен своим присутствием около весового модуля. Из-за чего процесс взвешивания затрудняется. В результате чего, было принято решение разработать мобильное приложение для контроля весовым модулем. При этом оператор нахо-

дится в непосредственной близости и видимости весовой ж/д платформы.

Программно-аппаратный комплекс взаимодействия ж/д вагонов состоит из следующих компонентов:

- весовая платформа;
- весовой контроллер;
- информационное табло;
- ip-видео камеры;
- сетевой коммутатор;
- ПК оператора;
- мобильное устройство под управлением android;
- wi-fi роутер.

На рисунке 1 представлена схема взаимодействия разработанного программного средства с оборудованием системы взвешивания предприятия.

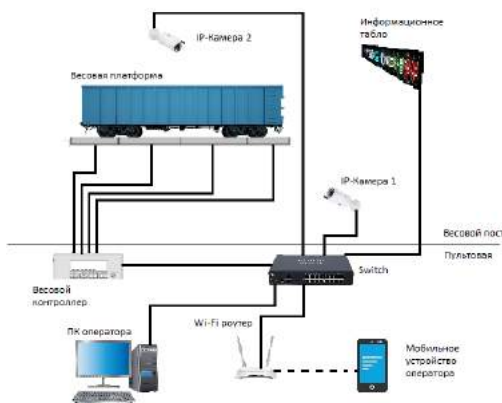


Рис. 1 – Схема взаимодействия программного средства с аппаратным комплексом предприятия

II. ФУНКЦИОНАЛ И АЛГОРИТМ РАБОТЫ МОБИЛЬНОГО ПРИЛОЖЕНИЯ

Было реализовано мобильное ПС обладающее следующими функциональными возможностями:

- подключение к весовому контроллеру по локальной сети с помощью TCP-протокола;
- взаимодействие с весовым контроллером посредством отправки запросов на сервер

и вывод ответов от сервера на графический интерфейс приложения;

- ввод настроек для доступа к серверу и к IP-камере;
- отображение видеопотока с IP-камеры;
- создание скриншотов видеопотока;
- сохранение состояния весов в статическом режиме в базе данных SQLite;
- сохранение скриншотов в локальном хранилище;
- формирование отчета о взвешивании за выбранную дату в pdf-файле и отправка этого файла по всем предоставленным приложениям.

Алгоритм работы мобильного приложения:

- в начале работы, оператор входит в настройки мобильного приложения, чтобы создать весовой модуль и заполнить все необходимые поля, указанные в настройках для подключения к весовому контроллеру и IP-камерам;
- далее, оператору нужно зайти на главную страницу приложения и нажать на кнопку подключиться в меню приложения. Начнется процесс подключения к весовому модулю, а чем будет проинформирован оператор из информационного поля приложения;
- после чего, на информационном поле приложения будет выведено меню весового контроллера. Взаимодействие с весовым контроллером осуществляется по нажатию на кнопки, которые находятся на главной странице приложения. По нажатию на кнопку отправляется запрос на весовой модуль по интерфейсу WI-FI с помощью протоколов TCP/IP, после чего весовой модуль отправляет ответ, который обрабатывается приложением;
- при входе в статический режим взвешивания на главной странице приложения появляется кнопка сохранить, при нажатии на которую сохраняется текущий вес груза вагона и формируется снимок экрана с IP-камеры, что оператор сможет увидеть в таблице, которая также находится на главной странице и отображает сохраненные протоколы за текущий день;
- для того, чтобы оператор сформировал отчет по взвешиваниям, ему необходимо войти на страницу «Протоколы», выбрать дату за какой день он хочет сформировать отчет и нажать кнопку «Отправить протокол», после чего приложение сформирует pdf-файл и отобразит все возможные при-

ложения, по которым можно выслать этот файл.

Примеры работы мобильного приложения представлены на рисунке 2.



Рис. 2 – Примеры работы мобильного ПС

Данное программное средство разработано на языке Kotlin в среде разработки Android Studio [3]. Для работы с видеорядом использована библиотека libvlc [4]. Взаимодействие с весовым контроллером осуществляется посредством библиотеки java.net.Socket. Формирование снимка экрана из IP-камеры осуществляется библиотекой FFmpegAndroid [5].

III. ЗАКЛЮЧЕНИЕ

Разработанное мобильное приложение позволяет ускорить и облегчить процесс взвешивания оператору подвижных составов в статическом режиме с обработкой полученных данных и автоматического формирования отчета на основе этих данных, что повышает эффективность процесса технологического учета грузов на предприятиях.

IV. СПИСОК ЛИТЕРАТУРЫ

1. Деменковец, Д. В. Алгоритм автоматического взвешивания железнодорожных вагонов / Д. В. Деменковец, К. Ю. Дубицкая // Компьютерные системы и сети: 55-я юбилейная научная конференция аспирантов, магистрантов и студентов, Минск, 22-26 апреля 2019 г. / БГУИР. – Минск, 2019. – 94–95 с.
2. Деменковец, Д. В. Система автоматической фото, видеофиксации и записи процесса взвешивания железнодорожных вагонов / Д. В. Деменковец // Информационные технологии и системы 2017 (ИТС 2017): Материалы международной научной конференции, БГУИР, Минск, Беларусь, 25 октября 2017 / БГУИР. – Минск, 2018. – 22–23 с.
3. Kotlin Programming Language [Электронный ресурс] / Режим доступа: <https://kotlinlang.org/docs/home.html>.
4. VideoLAN [Электронный ресурс] / Режим доступа: <https://www.videolan.org/vlc/libvlc.html>.
5. FFmpegAndroid [Электронный ресурс] / Режим доступа: <https://github.com/tanersener/mobile-ffmpeg>.

ОСОБЕННОСТИ ОРГАНИЗАЦИИ ПРИЕМНОЙ КАМПАНИИ В ВУЗАХ РЯДА СТРАН СНГ

Кисель Т. В.

Кафедра информационных технологий и интеллектуальных систем,
Полесский государственный университет
Пинск, Республика Беларусь
E-mail: kisel_t@mail.ru

В рамках данной статьи рассматриваются особенности организации и проведения вступительной кампании в вузах бывших стран Советского Союза: Беларуси, России, Азербайджана, Казахстана и Кыргызстана. Особое внимание акцентируется на уровне автоматизации отдельных процессов вступительной кампании в каждой из вышеперечисленных стран.

ВВЕДЕНИЕ

Изучение национального опыта, по части организации приемной кампании, включает ряд вопросов, таких как: основная форма вступительных испытаний в вузах; организация ответственная за их проведение; возможность подачи документов параллельно в несколько вузов; на несколько специальностей одновременно; способы подачи документов и т.п. Ответы на поставленные вопросы позволят сравнить уровень организации вступительной кампании в вузах вышеперечисленных стран.

I. РОССИЙСКАЯ ФЕДЕРАЦИЯ

Основной формой вступительных испытаний в вузах является единый государственный экзамен (ЕГЭ), который организуется и проводится Федеральной службой по надзору в сфере образования и науки [1].

Максимальное количество учебных заведений, в которые разрешается параллельно подавать документы – равно 5. Количество специальностей, по которым абитуриент может одновременно участвовать в конкурсе в конкретном учебном заведении – от 2 до 10 на усмотрение вуза.

Документы в вуз, можно подать одним из следующих способов: лично; через операторов почтовой связи; в электронном виде посредством электронной информационной системы вуза (сайт либо электронная почта); в электронном виде с использованием портала государственных услуг, через суперсервис «Поступление в вуз онлайн» [2].

Подавать можно как оригиналы, так и копии документов, не дожидаясь результатов ЕГЭ. При заполнении заявления достаточно только указать предметы, которые абитуриент собирается сдавать. После публикации результатов экзаменов, полученные баллы автоматически будут переданы в вуз.

Проведение конкурсного отбора и зачисление осуществляется приемными комиссиями вузов.

II. АЗЕРБАЙДЖАНСКАЯ РЕСПУБЛИКА

Прием в вузы Азербайджана, от момента регистрации для сдачи вступительных испытаний, до зачисления, осуществляется централизованно в рамках всей республики Государственным экзаменационным центром (ГЭЦ)[1]. Для сдачи вступительных экзаменов абитуриент создает на сайте ГЭЦ личный кабинет, заполняет электронное заявление, подает документы в электронном виде.

После проведения вступительных испытаний и объявления результатов, начинается этап отбора специальностей для участия в конкурсе. Отбор специальностей осуществляется в электронном виде через официальный сайт центра. Абитуриенты могут выбрать до 15 специальностей различных учебных заведений, в желаемой последовательности сформировав электронную заявку.

После проведения конкурсного отбора и объявления результатов, абитуриенты, ставшие студентами, должны зарегистрироваться в учебном заведении, в которое они поступили. Регистрация осуществляется онлайн, на сайте электронного правительства (portal.edu.az). Отказ от регистрации в указанный срок рассматривается как отказ поступающего от зачисления.

III. РЕСПУБЛИКА КАЗАХСТАН

Основная форма вступительных экзаменов в вузы – единое национальное тестирование (ЕНТ). Организация ответственная за проведение вступительных испытаний – Национальный центр тестирования [1]. Заявку для участия в ЕНТ абитуриенты подают онлайн на сайте центра.

После объявления результатов ЕНТ, абитуриент подает заявку для участия в конкурсе на присуждение образовательного гранта. Грант предусматривает возможность получения высшего образования за счет государственных средств. Подать заявку и документы в электронном виде можно через веб-портал «электронного правительства» (www.egov.kz) либо виртуальную приемную комиссию вуза (одного из вы-

бренных). Во всех вузах, запущены виртуальные приемные комиссии, для оказания абитуриентам онлайн-консультации при подаче документов на конкурс. В заявке абитуриент указывает выбранную им группу образовательных программ (ГОП) и учебное заведение. При совпадении профильных предметов, абитуриент может выбрать до четырех ГОП и до четырех вузов.

Проведение конкурса по присуждению образовательных грантов осуществляет Республиканская конкурсная комиссия, централизованно по всей республике. Образовательные гранты присуждаются на конкурсной основе в соответствии с баллами сертификатов ЕНТ согласно очередности ГОП, заявленных поступающим, с выдачей электронного свидетельства о присуждении образовательного гранта. Лица, получившие свидетельство о присуждении образовательного гранта, подают заявление о приеме в указанный в свидетельстве вуз, через виртуальную приемную комиссию вуза либо веб-портал «электронного правительства».

IV. КЫРГЫЗСКАЯ РЕСПУБЛИКА

Основная форма вступительных испытаний в вузах Кыргызстана – общереспубликанское тестирование (ОРТ) [1]. Тест проводится независимой тестовой службой в тесном сотрудничестве с Министерством образования и науки Кыргызской Республики. Регистрация на ОРТ проходит в онлайн-режиме на сайте ведомства.

Прием и оформление документов, проведение конкурсного отбора и зачисление, осуществляется централизованно через единую республиканскую автоматизированную информационную систему «Абитуриент Online».

Процедура зачисления предусматривает наличие трех туров. Если после первого тура в вузе остаются вакантные места, то вуз проводит второй тур. Правила конкурсного зачисления во всех турах одинаковы. В каждом туре абитуриент, претендующий на получение образовательного гранта, вправе выбрать один вуз и одну специальность.

Для участия в конкурсе абитуриент подает заявку через автоматизированную информационную систему «Абитуриент Online». На сайте размещены все учебные заведения Кыргызстана, факультеты и специальности. Абитуриенту нужно определиться с выбором вуза и подать электронный талон ОРТ.

Абитуриент, прошедший по конкурсу, должен в трехдневный срок, подтвердить свое желание учиться на данной специальности, подав в вуз пакет документов в электронном виде через АИС «Абитуриент Online». Оригиналы документов предоставляются в вуз в начале учебного года.

V. РЕСПУБЛИКА БЕЛАРУСЬ

Основной формой вступительных испытаний в вузы является централизованное тестирование (ЦТ), которое организует и проводит Республиканский институт контроля знаний (РИКЗ). Заявление для участия в ЦТ можно подать лично, посетив один из пунктов регистрации либо через систему предварительной регистрации на сайте РИКЗ (чтобы забрать пропуск на ЦТ, все равно придется лично посетить пункт регистрации).

После получения на руки сертификатов ЦТ, абитуриенты лично подают оригиналы документов и только в один вуз. На сегодняшний день, прием документов, проведение конкурсного отбора и зачисление осуществляется приемными комиссиями многих вузов практически «в ручном режиме». Хотя в Белорусском государственном университете информатики и радиоэлектроники (БГУИР) уже около 10 лет используется автоматизированная система подачи заявлений и зачисления (АСПЗиЗ). Создание системы было поручено БГУИР Министерством образования еще в 2010 году. Данную разработку планировалось в перспективе внедрять в масштабах республики. Однако по ряду причин, как технических, так и организационных, а также отсутствия нормативной базы, вопрос остается открытым.

ЗАКЛЮЧЕНИЕ

Вступительная кампания в вузах Республики Беларусь в нынешнем году проходила с особенностями (нововведения коснулись технических аспектов подачи документов). Одна из ее особенностей – предварительная электронная регистрация (изменения направлены на оперативную подачу документов, сокращение времени пребывания абитуриента в помещениях приемной комиссии).

Однако, несмотря на предпринимаемые попытки, механизм организации и проведения вступительной кампании в вузах нашей республики, по сравнению со странами постсоветского пространства, требует совершенствования.

СПИСОК ЛИТЕРАТУРЫ

1. Болотов, В. А. Поступление в вузы в постсоветских странах: экзамены как инструмент решения государственных задач [Текст]: сост. и науч. ред. В. А. Болотов, Р. В. Горбовский; Нац. исслед. ун-т «Высшая школа экономики». - М.: Изд. дом Высшей школы экономики, 2020. - 168 с.
2. Публичная витрина Суперсервиса «Поступление в ВУЗ онлайн» [Электронный ресурс]. - Режим доступа: <https://superserviceinfo.bmstu.ru/> - Дата доступа: 20.10.2021

СОЗДАНИЕ ДИНАМИЧЕСКОЙ МОДЕЛИ ЭЛЕКТРОМЕХАНИЧЕСКОЙ СИСТЕМЫ С ИСПОЛЬЗОВАНИЕМ MATLAB И AUTOCAD

Клютко М. В., Кобринец В. П., Карпович Д. С.
Кафедра автоматизации производственных процессов и электротехники,
Белорусский государственный технологический университет
Минск, Республика Беларусь
E-mail: m.kliutko@gmail.com, kobrinets@rambler.ru, karpovich@tut.by

В данной статье представлена графическая электромеханическая система (ЭС) с 3D-моделированием и симуляцией манипулятора с 5-тью степенями свободы. Динамическая модель ЭС разработана с помощью программного обеспечения Matlab SimMechanics и AutoCAD. В разработанной модели используются условные габариты манипулятора. Цель нашей работы в том, чтобы этот человеко-машинный интерфейс будет использоваться для предварительного тестирования другого типа управления перед применением к реальной электромеханической системе.

ВВЕДЕНИЕ

Робототехника – отрасль машиностроения, занимающаяся разработкой, созданием, эксплуатацией машин и устройств, запрограммированных на самостоятельное выполнение конкретных задач.

Электромеханическая система – это машина, которая может манипулировать объектами и выполнять различные движения, продиктованные легко изменяемым программ. ЭС, оснащенные устройствами искусственного интеллекта могут справиться с неожиданными и новыми сложными ситуациями. Они в основном используются в промышленности для выполнения повторяющихся манипуляции, особенно когда производственный процесс подвержен регулярным изменениям [1,2].

Преимущество электромеханической системы (робота-манипулятора) от человека – это его постоянство: он может выполнять одно и то же движение тысячи раз подряд без чувств усталости. Второе преимущество, ЭС могут быть сконструированы так, чтобы противостоять условию, которое может быть вредным или смертельным для человека [3].

По этой причине наша работа состоит в том, чтобы смоделировать электромеханическую систему, используя человеко-машинный интерфейс.

I. КИНЕМАТИЧЕСКАЯ СХЕМА ЭС

Электромеханическая система с 5-тью степенями свободы имеет компактную форму, есть возможность установки на пол, подвешивание к потолку или установку на стены. Данный манипулятор демонстрирует отличную свободу движений в любых ситуациях, идеально подходит для сложных задач, таких как сборка, погрузка и разгрузка небольших деталей.

Нами рассмотренная электромеханическая система состоит из элементов: основание, плечо, локоть, предплечье и запястье.

Две основные оси и три оси запястья обеспечивают 5-осевое перемещение (см. рис. 1).

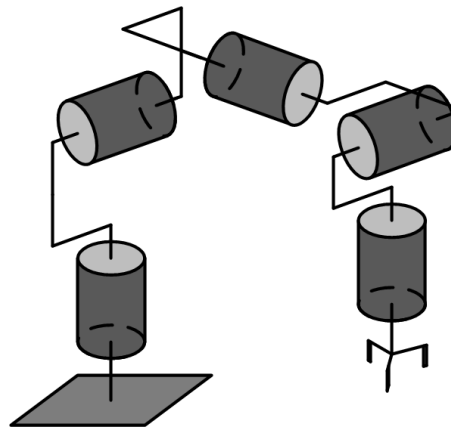


Рис. 1 – Кинематическая схема ЭС в рабочем состоянии

Идеально подходит для различных рабочих пространств: напольное, настенное или потолочное крепление. Использование энкодеров устраняет необходимость обнулять агрегат при включении.

II. МОДЕЛЬ ЭС В AUTOCAD

Система 3D моделирования в AutoCAD – полнофункциональное решение для трехмерного моделирования, включающее работу с твердыми телами, поверхностями, сеточными телами, среду визуализации, параметрическое проектирование.

3D модель манипулятора можно реализовать в различных программах. Мы выбрали AutoCAD, так как это одно из удобных и распространенных программных обеспечений, которой пользуются студенты, дизайнеры, инженеры и другие специалисты различных профессий.

III. ДИНАМИЧЕСКАЯ МОДЕЛЬ ЭС В MATLAB

Для создания динамической модели в Matlab-Simulink использовались блоки

SimMechanics. Библиотека блоков предоставила нам инструменты для формулирования и решения движений полной электромеханической системы [4].

Схема для реализации динамической модели ЭС с 5-тью степенями свободы изображена на рис. 2.

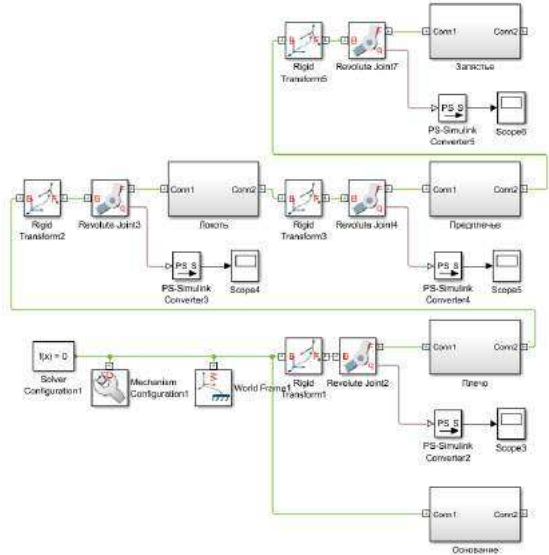


Рис. 2 – Схема для реализации динамической модели ЭС с 5-тью степенями свободы

Запустив данную схему (см. рис. 2) мы получаем динамическую модель нашего манипулятора, который изображен в окне Mechanics Explorers (см. рис. 3).

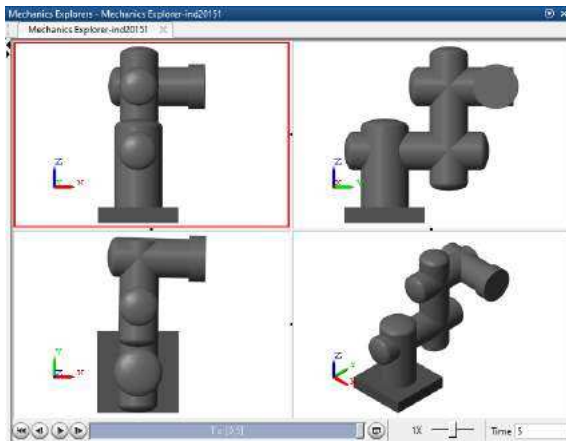


Рис. 3 – Динамическая модель ЭС

Каждый сустав манипулятора отличается визуально и по габаритам, ниже (см. рис. 4, 5) представлен четвертый элемент и основание ЭС.

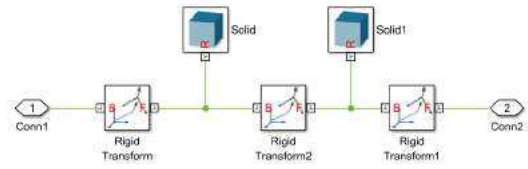


Рис. 4 – Схема для реализации четвертого сустава электромеханической системы (предплечье)

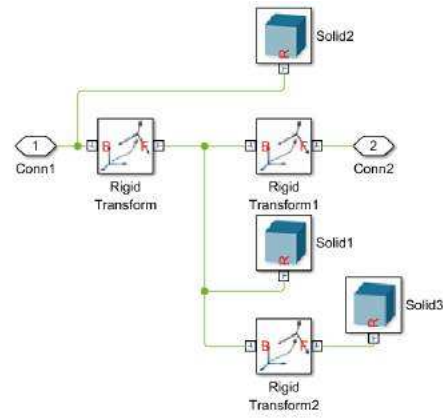


Рис. 5 – Схема для реализации первого сустава электромеханической системы (основание)

IV. ЗАКЛЮЧЕНИЕ

Проделанная работа направлена на изучение и 3D моделирование манипулятора, чтобы упростить взаимодействие человека с электромеханическими системами. Можно выделить, что наша работа может быть использована для управления всевозможных манипуляторов с реальными ограничениями.

Представили кинематическую схему и динамическую модель ЭС. Эти модели задействованы для моделирования ЭС с использованием Matlab.

Объемная модель системы отражает все основные динамические свойства объекта и позволяет на своей основе создавать и моделировать движения и различные режимы работы.

1. Колесов, Ю. Б. Моделирование систем. Динамические и гибридные системы / Ю. Б. Колесов, Ю. Б. Сениченков // – СПб.: БХВ-Петербург, 2006. – 224 с.
2. Козлов, О. С. Программный комплекс для исследования динамики и проектирования технических систем / О. С. Козлов, Д. Е. Кондаков, Л. М. Скворцов [и др.] // Информационные технологии. – 2005. – № 9.
3. Lallemand, J. P. Robotics. Fundamental aspects / J. P. Lallemand, S. Zeghloul // Paris. – 1994. – P. 312.
4. Дьяконов, В. П. SIMULINK 4. Специальный справочник / В. П. Дьяконов // – СПб.: Питер, 2002. – 532 с.

СИСТЕМЫ ВЕРИФИКАЦИИ СУБЪЕКТОВ ДОСТУПА НА ОСНОВЕ РЕЧЕВОГО СИГНАЛА

Крищеневич В.А., Захарьев В.А.
Кафедра защиты информации, Кафедра систем управления,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: {krish, zahariev}@bsuir.by

Доклад посвящен анализу предметной области связанной с построением систем верификации пользователей по биометрическим признакам (в частности, на основе характерных особенностей голоса) для управления доступом к персональной информации. Классификации подобных систем по типу используемой информации, анализу основных методов и алгоритмов, применяемых при построении современных систем данного типа.

ВВЕДЕНИЕ

В настоящее время, в связи с широким распространением персональной вычислительной техники в различных областях жизни, особо остро встает вопрос ограничения доступа пользователя к информации, хранящейся на данных устройствах. Для этих целей, широко применяются различные методы идентификации и верификации субъектов доступа. При этом все чаще стандартные парольные системы защиты, персональные идентификаторы и удостоверения личности заменяются или дополняются биометрическими системами идентификации и верификации пользователей. Это связано с тем, что наличие стандартных механизмов, зачастую является не достаточным для обеспечения необходимого уровня надежности защиты информации. Поэтому при создании современных систем контроля и управления доступом, применяются методы биометрического распознавания, основанные на вычислении и анализе индивидуальных характеристик субъекта доступа. На текущий момент в качестве измеряемых параметров используют различные человеческие черты, такие как отпечатки пальцев, изображения лица, записи речи диктора и т.д. Примерами таких систем могут выступать хорошо известные и зарекомендовавшие себя технологии и системы применяемые в персональных устройствах, такие как TouchID, FaceID и т.д.

I. ВИДЫ ИНФОРМАЦИИ ПРИМЕНЯЕМЫЕ ДЛЯ ВЕРИФИКАЦИИ

В процессе распознавания речи могут использоваться лингвистические (непосредственно связанные с конкретным языком) и паралингвистические (не связанные с конкретным языком) особенности речи говорящего. К лингвистическим особенностям, относятся параметры характеризующие само сообщение, их определение позволяет ответить на следующие вопросы:

- О чем говорят?
- На каком языке говорят?

Паралингвистические особенности, отражают индивидуальные особенности и эмоционально-физиологические характеристики речи говорящего и в большой степени характеризуют источник речевого сообщения, т.е. диктора. Их обработка и анализ в системе позволяет ответить на следующие основные вопросы:

- Кто говорит?
- Пол говорящего?
- Возраст говорящего?
- Эмоциональное состояние говорящего?

Для верификации диктора используются признаки и информация второго вида, т.е. относящаяся к паралингвистической.

II. КЛАССИФИКАЦИЯ СИСТЕМ

Для понимания особенностей функционирования необходимо дать классификацию основных типов систем, которые относятся к системам распознавания диктора, поскольку системы верификации диктора являются одним из подклассов подобных систем. Таким образом системы распознавания диктора, в зависимости от основных процессов происходящих в них, можно разделить на системы:

- Идентификации: определение личности неизвестного среди группы записанных голосов. Звукозапись неизвестного сравнивается с моделями всех зарегистрированных голосов и определяются вероятности их соответствия.
- Верификации: подтверждение заявленной личности человека с помощью речевого сигнала. Сравнивается предоставленный образец записи голоса с имеющимися образцами и на основании степени соответствия принимается решение принять или отклонить возможность выполнения дальнейших действий в системе.
- Диаризации: разделение входного аудиопотока на сегменты в соответствии с личностью диктора, чей голос звучит в данный момент времени.

III. АЛГОРИТМ ВЕРИФИКАЦИИ ДИКТОРА

Верификация диктора направлена на определение того, соответствует ли личность диктора заявленной, и обычно требует одного сравнения. С другой стороны, идентификация дикторов среди N-размерной группы требует N число сравнений.



Рис. 1 – Обобщенный алгоритм верификации диктора

Основные этапы работы системы в процессе выполнения алгоритма, включают в себя следующие основные шаги:

- анализ и извлечение из речевого сигнала признаков характеризующих голос диктора (модуль обработки сигнала);
- классификация полученных признаков с помощью моделью диктора, хранящейся в системе (модуль обработки признаков);
- принятие решения об успешности верификации на основании результатов сравнения.

Для выполнения процесса классификации требуется следующих действий:

- Разработка универсальной модели: проанализировать сходства в параметрах голосов дикторов. Сама модель именуется универсальной фоновой моделью (УФМ), поскольку является инвариантной по отношению к конкретным фонемам языка. Она может быть определена с учетом пола, т.е. есть отдельная модель для мужских и женских голосов, но является общей для всех дикторов.
- Разработка модели диктора: проанализировать различия в параметрах голосов диктора. На этом этапе обычно создается одна модель для каждого рассматриваемого уникального диктора.
- Верификация: голос субъекта сравнивается с ранее зарегистрированными голосами, согласно алгоритму представленному выше.

Основные проблемы, возникающие при построении систем верификации:

- большое количество данных, необходимых для обучения моделей;
- несоответствие между данными на этапе обучения и этапе верификации (функционирования систем), поскольку могут измениться параметры акустического окружения (посторонние шумы).

Разница между вводными данными и проверочными данными может быть связана с:

- аппаратными средствами (микрофон);
- окружающей средой (шумы и помехи);
- каналом передачи (стационарный, VoIP);
- самим диктором (состояние здоровья).

Процесс принятия решения в системах верификации диктора осуществляется на основе определения вероятности того, что голос диктора соответствует уникальной модели диктора, и вероятности того, что характеристики голоса диктора относятся к универсальной фоновой модели.



Рис. 2 – Применение универсальной фоновой модели для верификации диктора

Далее определяется отношение данных вероятностей. Если соотношение выше порога принятия решений, то система принимает решение, что образец принадлежит конкретному диктору. В противном случае принимается, считается что диктор не определен. Стоит обратить внимание, что УФМ решение выносится с учетом особенностей голоса характерных для дикторов конкретного пола.

ЗАКЛЮЧЕНИЕ

В докладе рассмотрены вопросы, связанные с системами верификации диктора на основе речевого сигнала: какие виды информации применяются для верификации, классификация подобных систем, обобщенный алгоритм функционирования систем, применение унифицированной фоновой модели для осуществления процесса верификации.

Технология биометрической верификации по голосу представляет особый интерес при её использовании в составе мобильных устройств, поскольку позволяет проводить процесс верификации без необходимости прямого тактильного или визуального взаимодействия с устройством, что в значительной степени улучшает показатели удобства использования подобных систем.

СПИСОК ЛИТЕРАТУРЫ

1. Mak M.W., Jen-Tzung C. Machine Learning for Speaker Recognition. Cambridge University Press, 2020 – 336 p.
2. Fabien M., Basics of Speaker Verification. [Электронный ресурс]. Режим доступа: https://maelfabien.github.io/machinelearning/basics_speech. Дата доступа: 15.10.2021.

ОБРАБОТКА ВИДЕОПОТОКА В СИСТЕМЕ ТЕХНИЧЕСКОГО ЗРЕНИЯ

Кузнецов А. П., Снисаренко С. В.

Кафедра систем управления,

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: kafsu@bsuir.by

Представлен метод обнаружения объектов в зашумленном видеопотоке с высоким FPS в системе технического зрения с использованием библиотек: OpenCV для работы с изображениями и видео, NumPy для проведения операций над матричным представлением фреймов, Numba для ускорения части операций, которые выполняются в среде Python.

ВВЕДЕНИЕ

Системы технического зрения для обработки видеoinформации снабжаются необходимыми алгоритмами и программным обеспечением. Они могут иметь наборы прикладных программ реального времени для различных технологических задач с микропроцессорной обработкой информации. Тогда они чисто программным путем быстро перенастраиваются для обслуживания тех или иных технологических операций. Следовательно, совместно с роботами, для оцувствления которых системы технического зрения служат, и с другим оборудованием они представляют единый многофункциональный роботизированный технологический комплекс, способный автоматически работать в различных ситуациях как при смене вида операций, так и в не вполне определенной или меняющейся обстановке. Программное обеспечение в системах технического зрения зависит от алгоритмов обработки информации, т. е. от способов идентификации предметов и выбранных признаков для этого, а эти факторы существенно влияют на обеспечение требования работы системы в реальном времени. Детекция движения – актуальная проблема в решении задач видеоаналитики. Часто она осуществляется за счет сравнения изменяемой части изображения с неизменяемой, что позволяет выделить фон и движущийся объект. Точность выделения контура движущегося объекта зависит от точности выделения области движущегося объекта. Выделяют такие ошибки выделения области движущегося объекта, как шум, ложное обнаружение (ошибка первого рода) и пропуск цели (ошибка второго рода). Для решения данных проблем в основном применяют метод соответствующей фильтрации на стадии обнаружения.[1]

I. ПОСТАНОВКА ЗАДАЧИ

Для интеллектуальной системы обработки видеопотока, полученного с камеры технического зрения робота, необходимо реализовать программно алгоритм идентификации движения объекта, используя базовый метод определения

движения. За базовый принят алгоритм вычитания фона с использованием смеси Гауссовых распределений. В основе базового алгоритма лежит усреднение, матричное вычитание и отсеечение шума. Этапы алгоритма следующие:

1. Сжатие изображения для удаления лишних деталей с интерполяцией для лучшего усреднения;
2. Гауссовское размытие для удаления лишних деталей и снижения уровня зашумленности;
3. Вычитание из текущего фрейма фрейма фона; отсеечение “тусклых” фрагментов, которые характерны для фона; выравнивание цвета “ярких” фрагментов, которые характерны для движения;
4. Поиск контуров движущихся объектов.

Данный алгоритм имеет два недостатка, которые должны быть учтены при программной реализации – влияние шума и изменчивость фона. Общая схема последовательности обработки видеопотока в интеллектуальной системе технического зрения (см. рис. 1).



Рис. 1 – Последовательность обработки видеопотока в системе технического зрения

После восприятия информации в виде визуального изображения производится ее предварительная обработка для снижения посторонних помех, улучшения изображений отдельных элементов объекта или сцены, а затем она подвергается сегментации, заключающейся в подразделении сцены на составляющие части или элементы для выделения на изображении интересующих объектов. Последующее описание массива информации представляет собой определение характерных параметров, необходимых

для выделения требуемых объектов или элементов сцены и дальнейшего их распознавания посредством идентификации в соответствии с программным набором информации. И, наконец, посредством интерпретации окончательно устанавливается принадлежность "рассматриваемого" объекта к группе распознаваемых, установленных его зрительного образа.

Алгоритм будет реализован с применением библиотеки компьютерного зрения OpenCV, которую часто применяют в роботах для распознавания объектов окружающего мира.

II. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ОБРАБОТКИ ВИДЕОПОТОКА

При программной реализации алгоритма использовался следующий инструментарий Python: библиотека с открытым исходным кодом OpenCV для обработки видео, библиотека Numba для оптимизации ресурсоемких операций, библиотека NumPy для работы с матрицами. Рассмотрим этапы реализации.

Этап 1: Сжатие. На данном шаге происходит сжатие исходного изображения, что позволяет удалить мелкие детали и снизить уровень шума. Кроме того, это позволяет ускорить последующие операции и уменьшить количество потребляемой оперативной памяти.

Этап 2: Размытие. Уменьшенный фрейм подвергается гауссовскому размытию для того, чтобы еще более сгладить детали, выровнять фон, уменьшить шум и снизить влияние незначительных колебаний фона. Этап 3: Добавление в буфер движения. Циклический буфер движения используется для того, чтобы снизить уровень шума фона за счет усреднения. При этом в данном буфере новые кадры считаются более релевантными, чем старые. После того, как кадр вытесняется из буфера движения, он перемещается в циклический буфер фона. Размер буфера движения – от 3-5-10 фреймов (до 1/3 секунды). Чем больше кадров в буфере движения, тем выше вероятность обнаружения, но больше погрешность за счет включения в область движения "хвоста" объекта.

Этап 4: Добавление в буфер фона. Буфер фона – объемный буфер, в котором кадры формируют усредненную картину прошлого состояния среды за более продолжительное время, например, за 1 секунду для изменчивой среды, за 10 секунд для более стабильной среды. Чем больше кадров в истории буферного фона, тем меньший уровень шума создают краткосрочные изменения пространства, например, движение листьев, но дольше влияют объекты, которые продолжительное место находятся в кадре, например, высаживающая людей машина. Кадры, вытесняемые из буфера движения перемещаются в буфер фона, что позволяет задать "отставание" сцены фона от сцены движения.

Этап 5: Усреднение фона. Операция усреднения фона производится как среднее арифметическое среди кадров в буфере фона.

Этап 6: Усреднение движения. Операция усреднения движения производится в форме суммы кадров буфера движения с коэффициентами арифметической прогрессии, которая нормируется на сумму арифметической прогрессии. Это позволяет повысить значимость самых "свежих" кадров и понизить значимость старых кадров.

Этап 7: Вычитания фона из движения. На этом этапе производится определение мест, в которых фон отличается от анализируемого кадра, что соответствует движению. Результирующее изображение содержит ряд "цветных" пятен, которые соответствуют движущемуся предмету. На следующем изображении можно видеть результат данной операции (см. рис. 2).



Рис. 2 – Удаление фона из движения

Наблюдается "хвост" у движущихся объектов – это побочный эффект повышения стабильности за счет усреднения. т.к. выполняется перевод в оттенки серого цвета. Далее, отсекаются "тусклые" части изображения, которые характеризуют или "хвосты" или зашумления. Отсечение производится по порогу белого цвета. Чем выше порог отсечения, тем большее количество движений будет проигнорировано – в частности, при движении объектов, цвет которых слабо отличается от цвета фона, они будут игнорироваться.

Этап 8: Уменьшение размера кадра движения. Алгоритм построения контуров объектов зависит от площади "белого" в кадре. Кроме того, на седьмом этапе объекты могут содержать изъяны для отображения, могут наблюдаться мелкие движения, которые должны игнорироваться, что может вести к детекции "шума", ухудшению качества детекции, низкой производительности алгоритма. Для предотвращения этой ситуации необходимо дальнейшее уменьшение кадра.

Этап 9: Поиск областей, соответствующих движению и объединение пересекающихся областей. Алгоритм производит поиск в ширину по областям белого цвета, вычисляя охватывающие их рамки. Далее пересекающиеся рамки объединяются, чтобы уменьшить уровень дребезга.[2]

Рассмотрим особенности программной реализации. Чтобы обнаружить движение, необходимо вычислить разницу в значениях пикселей

двух изображений frame1 и frame2, предварительно их необходимо создать:

```
ret, frame1 = video.read()
ret, frame2 = video.read()
```

Для чтения видеопотока используется функция isOpened(), для этого организован следующий цикл:

```
while video.isOpened():
```

Далее необходимо вычислить абсолютную разницу между frame1 и frame2 с помощью метода cv2.absdiff():

```
difference = cv2.absdiff(frame1, frame2)
```

Далее необходимо перейти в оттенки серого с помощью метода cv2.cvtColor() т.к. обработка черно - белого изображения наиболее упрощена по сравнению с цветным и для идентификации движения объекта вполне применима:

```
gray = cv2.cvtColor(difference,
cv2.COLOR_BGR2GRAY)
```

Согласно алгоритму, следующий шаг - это размытие изображения для удаления шума. Для этого используется метод cv2.GaussianBlur(). Он принимает несколько аргументов: исходное изображение для размытия, выходное изображение, размер ядра Гаусса, стандартное отклонение ядра по оси x, стандартное отклонение ядра по оси y и тип границы.[3]

```
blur = cv2.GaussianBlur(gray, (5,5), 0)
```

Затем необходимо установить пороговое значение с помощью метода cv2.threshold(). Данный прием выделит движение, сегментируя фон и передний план (или движение). Метод cv2.threshold() принимает четыре аргумента: изображение, пороговое значение, максимальное значение, и тип порогового значения.

```
threshold = cv2.threshold(blur,
20, 255, cv2.THRESH_BINARY)
```

Далее используется метод cv2.dilate(), который принимает максимум 6 аргументов: изображение, ядро, привязку, итерации, тип границы и значение границы.

```
dilate = cv2.dilate (threshold, None,
iterations=3)
```

Метод cv2.findContours() находит контуры объекта. Он принимает три аргумента: исходное изображение, режим поиска и метод аппроксимации контура.

```
contour = cv2.findContours(dilate, cv2.RETR_TREE, v2.CHAIN_APPROX_SIMPLE)
```

Метод cv2.drawContours() отрисовывает контуры объекта. Его аргументы: изображение, контуры, contourIdx (это значение отрицательно, если нарисованы все контуры), цвет, толщина, тип линии, иерархия, максимальный уровень и смещение.

```
cv2.drawContours(frame1, contour, -1, (0, 0, 255), 2)
```

Отображается изображение с помощью метода cv2.imshow().

```
cv2.imshow("image frame1")
```

Затем необходимо установить изображение frame2 в качестве первого кадра и считывать видео для нового кадра, который помещается в параметр frame2.[4]

```
frame1 = frame2
ret, frame2 = video.read()
```

Выход из цикла можно реализовать по нажатию заданной клавиши. Если клавиша «v» нажата, произойдет выход из цикла:

```
if cv2.waitKey(40) == ord('v'):
break
video.release().
```

ЗАКЛЮЧЕНИЕ

Шум и изменение фона в видеопотоке влекут за собой микроизменения в изображении, которые возникают даже в отсутствии явного движения в кадре, в следствие чего становится проблематичной обработка видеоизображения на предмет поиска движения и идентификации объекта. Базовые алгоритмы не обрабатывают данные эффекты. Алгоритм, рассматриваемый в данной статье позволяет успешно справляться с последствиями шума, а именно представлена программная реализация алгоритма обнаружения движения в видеопотоке системы технического зрения робота с подавлением возникших шумов и влияния изменения фона. Использована библиотека алгоритмов компьютерного зрения и обработки изображений OpenCV для Python. Для увеличения скорости обработки видеопотока применена библиотека Numba, а для решения задач в рамках анализа данных с использованием матриц и векторов - библиотека NumPy.

III. СПИСОК ЛИТЕРАТУРЫ

1. Попов Е. П. Робототехника и гибкие производственные системы. / Е. П. Попов. - М.: Наука, 1987. - 190 с.
2. Высокопроизводительный детектор движения с подавлением шума на Python, OpenCV и Numba. [Электронный ресурс] / Bitworks software. - Томск, 2020. - Режим доступа: <https://bitworks.software/high-speed-movement-detector-opencv-numba-numpy-python.html>. - Дата доступа: 24.10.2021.
3. Шакирьянов Э. Д. Компьютерное зрение на Python. Первые шаги. /Э. Д. Шакирьянов. - Электрон. изд. - М. : Лаборатория знаний, 2021. - 163 с.
4. Fernández Villán, Alberto. Mastering OpenCV 4 with Python : A Practical Guide Covering Topics from Image Processing, Augmented Reality to Deep Learning with OpenCV 4 and Python 3. 7. / Fernández Villán, Alberto.- Packt Publishing Ltd, 2019. - 517 p.

ОЦЕНКА НАДЕЖНОСТИ ФАЗОСДВИГАЮЩЕГО ГЕНЕРАТОРА

Курулёв А. П.

Кафедра теоретических основ электротехники,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: alexrapakuru@yahoo.com

В статье рассматривается современный метод оценки показателей безотказности проектируемого фазосдвигающего генератора как важнейшей группы количественных показателей надежности.

ВВЕДЕНИЕ

Оценка показателей надежности радиоэлектронных устройств является обязательной процедурой, выполняемой на этапе проектирования аппаратуры. Актуальность задачи по расчету надежности объясняется тем, что она дает ответ на вопрос о целесообразности затрат, необходимых на обработку технологии и производство проектируемого устройства.

I. ПОКАЗАТЕЛИ НАДЕЖНОСТИ

Надежность определяет свойство устройства выполнять заданные функции в определенных условиях эксплуатации при сохранении значений основных параметров в заранее установленных пределах. Надежность зависит от количества и качества входящих в устройство элементов, от условий, в которых оно эксплуатируется (чем выше температура окружающей среды, чем больше относительная влажность воздуха, перегрузки при вибрации и т. д., тем меньше надежность).

Надежность характеризуется безотказностью, ремонтпригодностью, долговечностью, сохраняемостью и др.

Основу расчета показателей безотказности составляют математические модели прогнозирования эксплуатационной безотказности как элементов, так и устройства в целом. В моделях используются данные справочников по надежности, разработанных в Российской Федерации, а также справочников и стандартов по прогнозированию надежности электронного оборудования, изданных в США, Китае, Франции и Великобритании.

II. РАСЧЕТ НАДЕЖНОСТИ

Для расчета надежности схемы, приведенной на рис. 1, использованы следующие значения параметров: $U_1 = 200\text{мВ}$; $C_1 \div C_6 = 10\text{нФ}$; $R_1 \div R_6 = 10\text{кОм}$; $R_7 = R_8 = 200\text{кОм}$; $R_9 = R_{10} = 50\text{кОм}$

В таблице 1 приведены справочные и расчетные данные элементов схемы: интенсивность отказов $\lambda_i \cdot 10^{-6} \frac{1}{\text{ч}}$ коэффициент нагрузки $K_H = U/U_{max}$ поправочный коэффициент K_{II} , учитывающий K_H и температуру окружающей среды.

Справочные и расчетные данные.

Таблица 1 -

Элемент	λ_i	K_H	K_{II}	n
ИМС	0,028	0,35	5,5	1
Конденсатор	0,022	0,4	5	6
Резистор	0,044	0,4	3	10
Розетки	0,004	0,5	2,5	2

Суммарная эксплуатационная интенсивность отказов:

$$\lambda_{\Sigma} = \sum \lambda_i \cdot n_i = (0,028 \cdot 0,35 \cdot 5,5 + 0,022 \cdot 0,4 \cdot 5 \cdot 6 + 0,044 \cdot 0,4 \cdot 3 \cdot 10 + 0,004 \cdot 0,5 \cdot 2,5 \cdot 2) \cdot 10^{-6} = 8,56 \cdot 10^{-3} \frac{1}{\text{ч}}$$

Показатели безотказности фазосдвигающего генератора:

- наработка на отказ:

$$T_0 = \frac{1}{\lambda_{\Sigma}} = \frac{1}{8,56 \cdot 10^{-3}} = 11,5 \cdot 10^3 \text{ч.};$$

- вероятность безотказной работы за заданное время $t_3 = 1000\text{ч.}$:

$$P(t_3) = e^{-t_3 \cdot \lambda_{\Sigma}} = e^{-\frac{t_3}{T_0}} = 0,918;$$

- гамма-процентная наработка до отказа $\gamma = 99\%$:

$$T_{\gamma} = -\frac{\ln(\gamma/100)}{\lambda_{\Sigma}} = 117,39\text{ч.}$$

III. ЗАКЛЮЧЕНИЕ

Так как для схемы $P(t_3) = 0,918$, то 91,8% ее элементов должны работать безотказно в течении $t_3 = 1000$ часов.

Гамма-процентная наработка схемы до отказа при $\gamma = 99\%$ равна $T_{\gamma} = 117$ часов. Это означает, что у 99% элементов схемы в течении суммарной наработки, равной 117 часов, отказ не возникает. Таким образом, полученные результаты позволяют сделать вывод, что исследуемая схема приемлема при проектировании сложных устройств.

IV. СПИСОК ЛИТЕРАТУРЫ

1. Курулёв, А. П. Надежность изделий промышленной электроники. Минск, БГУИР, 2014.
2. Бородин, С. М. Обеспечение надежности при проектировании радиоэлектронных систем. Ульяновск, УлГТУ, 2010.
3. ГОСТ 27.301-95. Надежность в технике. Расчет надежности. Основные положения. М.: изд-во стандартов, 1997.
4. Дианов, В. Н. Диагностика и надежность автоматических систем. – М.: МГИУ, 2005.
5. Черкасов, Г. Н. Надежность аппаратно-программных комплектов. – СПб, СанктПетербург, 2005.

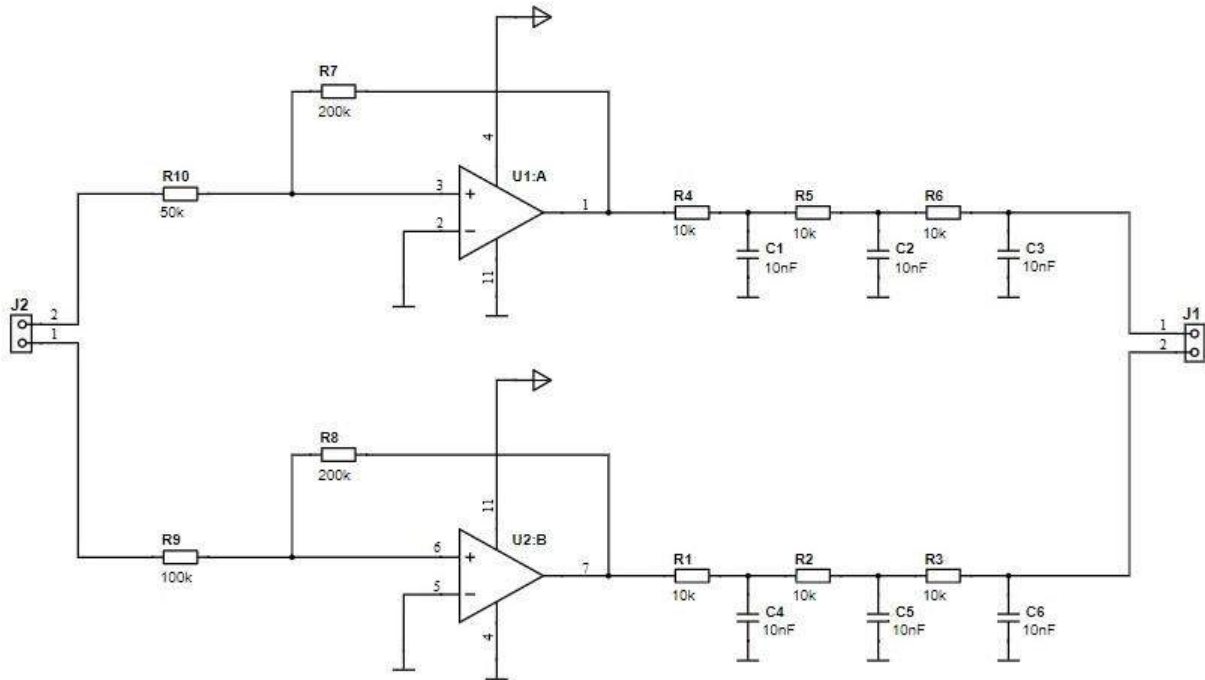


Рис. 1 – Схема фазосдвигающего генератора

ЭЛЕМЕНТЫ ВЫСШЕГО ПОРЯДКА В ПОЛОСОВЫХ ФИЛЬТРАХ

Петровский И. И., Свито И. Л., Шилин Л. Ю.

Кафедра теоретических основ электротехники,

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: Petrovski47@mail.ru, {svito, dekfitu}@bsuir.by

В работе рассматривается возможность применения элементов высшего порядка, предложенных как новые элементы в теории электрических цепей, в различных устройствах автоматики и других электротехнических системах. Рассматривается возможность применения этих элементов в фильтрующих устройствах, в частности, в полосовых фильтрах.

ВВЕДЕНИЕ

Известные методы преобразования L , C фильтров в полосовые фильтры имеют существенный недостаток, так как в продольной оси фильтра появляется колебательный контур. Этот контур трудно реализовать с помощью активных R , C элементов [1].

При применении элементов высшего порядка это можно существенно упростить, так как в продольной оси не требуется имитация индуктивности с помощью активных R , C элементов.

В дальнейшем будет показано, как через преобразование частоты с помощью элементов высшего порядка это может быть реализовано [3].

I. ЭЛЕМЕНТЫ ВЫСШЕГО ПОРЯДКА В ФИЛЬТРАХ

Известно, что трансформация частоты

$$p \sim \frac{1}{p}$$

преобразует фильтр нижних частот в фильтр высших частот. Если применить преобразование

$$p \sim \left(\frac{w_r}{\bar{p}} + \frac{\bar{p}}{w_r} \right)$$

то это даёт возможность перейти от фильтра нижних частот к полосовому фильтру. При этом

$$w_T = \frac{f_{+1} - f_{-1}}{f_{+1} * f_{-1}}$$

где f_{+1} верхняя граничная частота полосового фильтра, а f_{-1} – нижняя частота. Фильтр нижних частот с элементами высшего порядка имеет вид (рис. 1) [2].

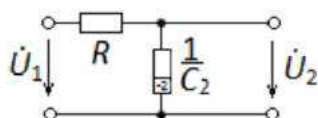


Рис. 1 – Фильтр нижних частот

После применения трансформации частоты полосовой фильтр представлен на рис. 2.

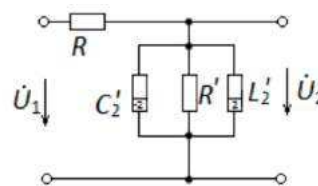


Рис. 2 – Полосовой фильтр

Элемент второго порядка преобразуется по формуле

$$\frac{1}{p^2 C^2} = \frac{1}{\frac{p^{-2} C^2}{w_T^2} + 2C_2 + \frac{C_2 W_T^2}{\bar{p}^2}}$$

Таким образом, при трансформации частоты возникает из одного элемента второго порядка три новых элемента, из них два элемента второго порядка и активное сопротивление. Так как в момент резонанса сопротивление параллельного участка полосового фильтра должно быть равно нулю, то

$$\frac{1}{p^2 L_2'} + \frac{1}{R'} + p^2 C_2' = 0$$

Решением уравнения

$$p^4 L_2' C_2' + p^2 \frac{L_2'}{R'} + 1 = 0$$

будет выражение

$$p_i = \pm \sqrt{-\frac{1}{2C_2' R'}} \pm \sqrt{\frac{1}{4(C_2')^2 (R')^2} - \frac{1}{L_2' C_2'}}$$

Так как этот параллельный участок фильтра рассматривается как резонансный контур, то не менее двух корней должны быть мнимыми. При этом, если рассматривать L_2' , C_2' , R' как положительные величины, то возможны три варианта.

1. Выражение во внутреннем квадратном корне для определения p_i может быть отрицательным. Тогда одна пара корней p_i будут иметь положительную действительную часть. Это недопустимо с точки зрения устойчивости.

2. Выражение во внутреннем квадратном корне равно нулю. Тогда получаются два двойных равных корня, что также недопустимо из-за устойчивости.
3. В работе исследуется случай, если

$$\frac{1}{4(C_2')^2(R')^2} - \frac{1}{L_2' C_2'} \geq 0.$$

Тогда неравенство

$$\left| \frac{1}{2C_2'R'} \right| > \left| \sqrt{\frac{1}{L_2'(C_2')^2(R')^2} - \frac{1}{L_2'C_2'}} \right|$$

всегда выполняется. Отсюда получаем две пары мнимых корней, которые отличаются друг от друга и лежат на мнимой оси. В этом случае система устойчива. Для определения элементов полосового фильтра действительна формула для C_2'

$$C_2' = \frac{C_2}{w_T^2}$$

а при заданном сопротивлении R' для L_2' действует ограничение

$$L_2' > \frac{4C_2(R')^2}{w_T}$$

II. РАСЧЁТ ЭЛЕМЕНТОВ ПОЛОСОВОГО ФИЛЬТРА И ОПРЕДЕЛЕНИЕ ХАРАКТЕРИСТИК

При практическом расчёте чаще всего применяется частотная трансформация

$$p \sim \frac{w_0}{w_b} \left(\frac{\bar{p}}{w_0} + \frac{w_0}{\bar{p}} \right).$$

где w_0 – средняя частота полосового фильтра; w_b – ширина полосы пропускания фильтра.

В этом случае сопротивление фильтра будет

$$Z(p) = \frac{1}{\frac{C_2}{w_b^2} p^2 + \frac{2w_0^2}{w_b^2} + \frac{w_0^4}{w_b^2 p^2} C_2}$$

Его график в нормированном виде представлен на рис. 3.

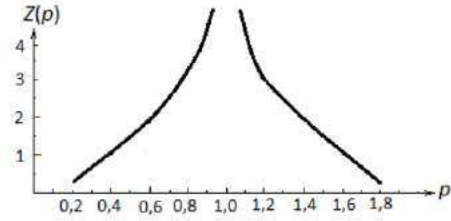


Рис. 3 – Операторное сопротивление фильтра

Передаточная функция фильтра тогда примет вид

$$W(p) = \frac{w_b^2 p^2}{RC_2 p^4 + (2w_0^2 C_2 R + w_b^2) p^2 + w_0^4 C_2 R}$$

В нормированном виде передаточная функция полосового фильтра представлена на рис. 4.

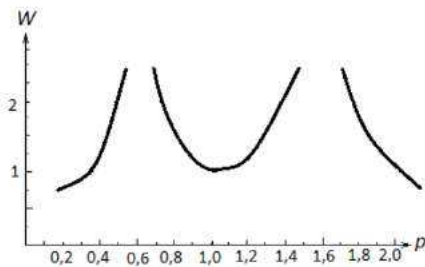


Рис. 4 – Передаточная функция

III. ЗАКЛЮЧЕНИЕ

Таким образом, из приведённого можно сделать вывод, что с помощью двух элементов высшего порядка можно получить любую передаточную функцию полосового фильтра и рассчитать его параметры.

IV. СПИСОК ЛИТЕРАТУРЫ

1. Panzer, K. Ein Beitrag zum Entwurf von aktiven Bandfiltern mit der minimalen Anzahl von Kondensatoren. Dissertation, TU Munchen, 1975.
2. Петровский И.И., Свито И.Л. Применение элементов высшего порядка в фильтрах нижних и высших частот. Материалы международной научной конференции «ИТС-2020», 2020 г., БГУИР Минск, Беларусь.
3. Philippow, E., Bruckner, P., Schaltungsanordnung zum Erzeugung sowie zum Transformation linearer und nichtlinearer frequenzabhandiger Zweitpole hoherer Ordnung. Patentanmeldung, TH Ilmenau, 1976.

SMART АЗС: ПОНЯТИЕ, СОСТОЯНИЕ, ПЕРСПЕКТИВЫ

Слинка К. И.

Кафедра компьютерной безопасности, Белорусский государственный университет

Минск, Республика Беларусь

E-mail: konstantinslinka@gmail.com

Данная работа раскрывает смысл понятия Smart АЗС и описывает принципы применения умных систем в работе АЗС. Разработана система управления АЗС, на базе которой можно построить энергоэффективную Автоматизированную Систему Контроля и Управления Энергообеспечением (АСКУЭ) АЗС. Такие технологии являются перспективными для создания полностью автономной АЗС.

ВВЕДЕНИЕ

В настоящее время наблюдается всеобщая тенденция к автоматизации и роботизации процессов. В связи с этим приоритет получают «Умные» технологии. В транспортной области одним из главных новшеств стало создание «умных» АЗС. На данный момент можно выделить несколько основных видов Smart АЗС: мини Smart АЗС – АЗС с элементами smart управления, Smart АЗС – АЗС с системой Smart, РАЗС – роботизированные АЗС.

I. ПРИНЦИПЫ РАБОТЫ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ

Принципы работы умных систем можно применять на АЗС, они основаны на ряде команд активация которых может быть осуществлена от запроса, направленного от человека, или от датчика. Связь между элементами системы поддерживается при помощи проводной или беспроводной связи. Вариант с проводными соединениями меньше подвергается поломкам и сбоям системы. Система может осуществлять контроль уровня освещения, отопительной и вентиляционной систем в зависимости от наличия посетителей, установить терминалы самообслуживания возле колонок и тепловые завесы. Выполнение данных принципов позволит осуществить экономию ресурсов и создавать комфортные условия обсуживаемых покупателей. Все это можно осуществить с помощью автоматизированной системы управления (АСУ). [1]

АСУ – комплекс аппаратных и программных средств, а также персонала, предназначенный для управления различными процессами в рамках технологического процесса, производства, предприятия. Важнейшая задача АСУ – повышение эффективности управления объектом на основе роста производительности труда и совершенствования методов планирования процесса управления. В общем случае, систему управления можно рассматривать в виде совокупности взаимосвязанных управленческих процессов и объектов. Обобщенной целью автоматизации управления является повышение эффективности использования потенциальных возможностей объекта управления.[2]

Наиболее вероятное направление развития Smart АЗС – это полная автономность. Для достижения автономности АЗС следует решить следующие проблемы: взаимодействие с клиентами, контроль работы оборудования и энергосбережение. Для взаимодействия клиента с АЗС без учета персонала необходимо оборудовать ее терминалами самообслуживания для покупки топлива. Для контроля работы оборудования необходимо оснастить АЗС "умной" системой. Для этого оборудование оснащается приборами электронного контроля на базе одноплатных компьютеров, подключенных к контроллеру с помощью беспроводного или проводного соединения. С добавлением элементов Smart управления можно оптимизировать энергопотребление систем освещения и контроля температуры за счет автоматизированной системы контроля и управления энергопотреблением (АСКУЭ).

II. ПОСТРОЕНИЕ МОДЕЛИ СИСТЕМЫ УПРАВЛЕНИЯ АСУ АЗС

В ходе данной работы были выработаны предложения по совершенствованию управления энергопотреблением АЗС с использованием современных smart-технологий. Для чего был проведен анализ ранее созданных моделей Smart АЗС, изучены внедренные технологии существующих Smart АЗС и построена модель структуры системы управления Smart АЗС.

Недостаточность ранее созданных моделей и методов для повышения эффективности сетей АЗС вызывает необходимость проведения системного анализа и разработки новых взаимосвязанных постановок задач, показателей, моделей, методов, алгоритмов и информационно-логических схем, образующих методологию рационального построения и непрерывного совершенствования структур сетей автозаправочных станций и эффективного автоматизированного управления процессами и объектами в данных системах.

Задачи анализа и синтеза структур систем управления связаны с понятием контур управления, которое вытекает из сущности управления, предполагающего наличие данных о состоянии объектов управления. Понятие «контур» и «управление» считаются связанными неразрывно,

а замкнутость контура управления – черта, присущая любой целеустремленной системе [3].

Построенная Модель Системы Управления (СУ) представляет собой трехмерную матрицу в ячейках которой находятся средства исполнения, выполняющие определенную элементарную задачу в пределах некоторого процесса, промежутка времени и функции управления. Построение варианта модели СУ проводилось по определенному алгоритму, который включает в себя: системный анализ объекта исследования, построение структуры управляемой системы, создание модели структуры системы управления, синтез структуры системы управления путем реализации процедур свертывания контуров управления и поиск варианта системы, оптимального или наилучшего по показателю эффективности предприятия. [4]

Управление представлялось в виде реализации функций управления S процессами P на периодах времени N средствами управления X_{pq} для достижения экстремального значения показателя эффективности или критерия K . Модель структуры системы управления задавалось декартовым произведением $F: S \times P \times N$. Элементы $F_{ijk} : C_i \times P_j \times N_k \times X_{pq}$ представляют собой модели элементарных задач управления. Контур управления определялся совокупностью элементарных задач по выполнению всех функций C_i . [5]

Не менее важной составляющей проекта является цифровая модель АЗС. Данная модель является имитацией работы АЗС с помощью программного обеспечения. Она способствует более быстрому обнаружению неполадок в оборудовании и ведению учета по расходу ресурсов. С этой моделью АЗС функционирует в цикле представленном на рисунке 1.

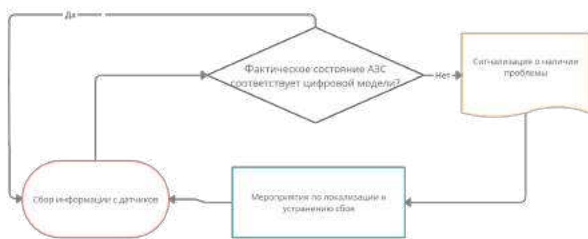


Рис. 1 – Цикл работы АЗС

Пилотным проектом в снижении эксплуатационных затрат и повышении эффективности работы являются АЗС Спартак и АЗС Озерная. За счет введения системы Smart АЗС и модернизации были получены следующие результаты:

1. Снижение энергопотребления АЗС на 5-15%. Снижение потребления достигается за счет контроля систем освещения, управления климатом и вентиляцией. В результате

нагрузка на климатическую технику упала на 54%, а на вентиляцию на 48%;

2. Повышение надежности АЗС. Про помощи цифровой модели АЗС стало проще выявлять место и причину поломки, а также реакция на такую поломку стала значительно быстрее.
3. Снижение нагрузки на персонал. При помощи статистических данных собранных цифровой моделью АЗС стало проще формировать отчеты по тратам и заказы топлива по мере необходимости. Так же эта система значительно уменьшает вероятность нарушения бухгалтерской отчетности.
4. Сбор данных. Данные полученные с цифровой модели можно удобно анализировать и использовать в дальнейшей оптимизации проекта.
5. Дистанционные проверка состояния и управление АЗС. Ответственное лицо способно проверить состояние и получить данные о работе АЗС практически в любом месте, при наличии выхода в интернет.

III. ЗАКЛЮЧЕНИЕ

Кроме энергосбережения перспективной технологией для Smart АЗС является взаимодействие с бортовыми компьютерами машин. Такое взаимодействие позволит водителям уделять меньше времени заправке автомобиля, так как в бортовом компьютере уже содержится вся информация необходимая для формирования заказа (уровень и марка топлива, а также его необходимое количество).

1. Что такое умный дом? [Электронный ресурс] / Платформа цифровой трансформации INTELVISION. СПб, 2006-2021. – Режим доступа: <https://www.intelvision.ru/blog/what-is-smarthome>. – Дата доступа: 18.04.2021.
2. Что такое АСУ? [Электронный ресурс] / Сайт компании Инженерные Сети. – Режим доступа: <https://as-system.ru/stati/chto-takoe-asu-vidy-avtomatizirovannykh-sistem-upravleniya>. – Дата доступа: 15.04.2021.
3. Акофф Р. О целеустремленных системах: пер. с англ. / Р. Акофф, Ф. Эмери; под ред. И с предисл. И.А. Ушакова. – 2-е изд., доп. – М.: Изд-во ЛКИ, 2008. – 272 с.
4. Безродный А. А. Модели структур и алгоритмы управления автозаправочными станциями / А. А. Безродный, А.Ф. Резчиков – Саратов: СГТУ, 2004. – 249 с.
5. Теоретико-множественное представление сложных систем, анализ, проектирование и синтез оптимальных (наилучших) структур: структуры систем управления, выделение системы из среды, многоконтурные системы, построение структур и выбор управляющих воздействий, потенциально-функциональная модель системы управления: курс для магистрантов по специальности «Прикладная физика»: учеб.-метод. пособие / БГУ; авт.-сост. Д.т.н., проф. кафедры «Интеллектуальные системы» БГУ Безродный А.А. – Минск, 2019. – 19 с.

ОЦЕНИВАНИЕ И КОМПЕНСАЦИЯ ПАРАМЕТРИЧЕСКИХ ВОЗМУЩЕНИЙ СЕРВОПРИВОДА

Хаджинов М. К., Павлова А. В.
Кафедра систем управления,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: kafsu@bsuir.by

Рассмотрен сервопривод с оцениванием влияния и компенсацией сигнальных и параметрических возмущений, действующих на объект управления. Предложена методика

ВВЕДЕНИЕ

В современных системах управления часто используются регуляторы в обратной связи, базирующиеся на наблюдателе. Система управления с наблюдателем обычно имеет контуры управления и оценивания. Наблюдатель формируется моделью объекта и регулятором контура оценивания. Обычно используется модель невозмущённого движения объекта [1]. После окончания переходного процесса в наблюдателе ошибка оценивания стремится к нулю при совпадении модели с объектом и отсутствии действующих на объект возмущений. Не нулевая ошибка оценивания свидетельствует о действии на объект сигнальных или параметрических возмущений.

На этом свойстве наблюдателей можно строить системы оценивания и компенсации возмущений обоих видов [2]. Проблема заключается в разделении оценок сигнального и параметрического возмущений, подаваемых на отдельные каналы компенсации возмущений. И если компенсация сигнальных возмущений достигается изменением сигнала управления, то компенсация параметрического возмущения требует перенастройки модели в наблюдателе и перенастройки регуляторов контуров управления и оценивания. Для выделения параметрических возмущений предлагается использовать пробные движения сервопривода с синхронным детектированием сигнала ошибки контура оценивания.

I. ЭТАПЫ УПРОЩЕНИЯ ЗАДАЧИ ИДЕНТИФИКАЦИИ ОБЪЕКТА УПРАВЛЕНИЯ

Параметрические возмущения в сервоприводе в первую очередь связаны с изменением момента инерции нагрузки, и в модели сервопривода могут быть сведены к изменению единственного коэффициента на входе интегратора ускорения. Сигнальные возмущения в сервоприводе это моментные возмущения, действующие на нагрузку, и инерциальные возмущения, связанные с перемещением самого сервопривода. Для простоты компенсации все сигнальные возмущения в модели сервопривода целесообразно привести к эквивалентному возмущению, действующему на вход [1]. Для выделения параметрических возмущений на фоне сигнальных пред-

лагается использовать гармонические пробные возмущающие движения как сервопривода, так и его модели с синхронным детектированием сигнала ошибки контура оценивания. Отфильтровав демодулированный сигнал ошибки контура оценивания, получим сигнал параметрического расхождения сервопривода и его модели. Остаётся подстроить коэффициент модели сервопривода под изменившийся сервопривод, а также при необходимости подстроить коэффициенты регуляторов контуров управления и оценивания. Подстройка коэффициентов регуляторов может осуществляться непрерывно или ступенчато при выходе параметрического возмущения за определённые пределы. В сервоприводе пробные сигналы для идентификации параметров возможно задавать на вход силового преобразователя. Однако специально создавать пробные сигналы не обязательно, так как в структуре силового преобразователя уже присутствуют сигналы естественной коммутации ШИМ регуляторов. Эти же сигналы следует использовать для синхронного детектирования сигнала ошибки контура оценивания. Таким образом будет достигнута максимально возможная частота пробных движений сервопривода, что положительно скажется на быстродействии контура оценивания параметров. Ускорить процесс оценивания параметрических возмущений можно за счёт упрощения контура оценивания путём установки датчика ускорения. Стандартный датчик координаты сервопривода позволяет построить контур оценивания наблюдателя. Для упрощения порядка модели на 2 единицы в контуре оценивания параметрических возмущений следует использовать датчик ускорения. Сигнал ошибки оценивания ускорения синхронно детектируется от пробных сигналов тока якоря электродвигателя и фильтруется интегратором. Сигнал интегратора служит для подстройки коэффициента преобразования тока, учитывающего изменение момента инерции нагрузки сервопривода, в модели наблюдателя и подстройки коэффициентов модальных регуляторов контуров управления и оценивания. Компенсатор сигнальных возмущений также может базироваться на сигналах датчика ускорений и представлять собой

интегратор сигнала ошибки оценивания ускорения. В этом случае порядок основного контура оценивания возмущения также уменьшится на 2 единицы, что упростит его настройку на параметрические возмущения. Для упрощения модели в контуре оценивания параметрических возмущений пробные движения можно рассматривать как задаваемые не напряжением ШИМ силового преобразователя, а током якоря электродвигателя, аналоговый сигнал которого в силовом преобразователе уже имеется. В этом случае из контура оценивания исчезают инерционности силового преобразователя и электромагнитных процессов электродвигателя. Однако возникает проблема с выделением гармоники в изменении тока электродвигателя для управления синхронным детектированием сигналов ошибки контура оценивания. Решить эту проблему можно с помощью аналогового или цифрового следящего фильтра, настроенного на частоту силового преобразователя и отслеживающего фазу колебаний управляющего тока электродвигателя. Такой фильтр может быть реализован как аналоговый на операционных усилителях, а также, как цифровой фильтр с повышенной частотой квантования или как цифровой фильтр с упреждением с частотой квантования силового преобразователя [3].

II. ЗАКЛЮЧЕНИЕ

Переход к токовым пробным движениям и измерениям ускорения сервопривода сводит модель идентифицируемого объекта к безинерционному объекту с коэффициентом преобразования тока в угловое ускорение двигателя. Что и позволяет сделать контур оценивания парамет-

рических возмущений самым быстрым контуром в системе управления. Процесс подстройки модели для компенсации параметрических возмущений происходит одновременно с процессом оценивания и может проходить так же быстро. Задавать приемлемую плавную форму переходных процессов идентификации не составляет проблему. Для сохранения формы переходных характеристик процессов регулирования необходимо перестраивать регуляторы контура управления и изменять полосу пропускания системы управления, при этом невозможно сохранить длительность переходных характеристик при увеличении инерционности нагрузки.

III. СПИСОК ЛИТЕРАТУРЫ

1. Хаджинов, М. К. Оценка и компенсация апостериорного матожидания случайных возмущений в системе модального управления на основе эталонных моделей / М. К. Хаджинов, В. А. Шевелева // Информационные технологии и системы 2014 (ИТС 2014) : материалы международной научной конференции, БГУИР, Минск, Беларусь, 29 октября 2014 г. / редкол. : Л. Ю. Шилин [и др.]. – Минск : БГУИР, 2014. – 352 с.
2. Хаджинов М. К. Компенсация сигнальных и параметрических возмущений на основе анализа ошибки контуров оценивания / М. К. Хаджинов, А.Т.Доманов, А.В.Павлова // Информационные технологии и системы 2015 (ИТС 2015): материалы международной научной конференции, БГУИР, Минск, Беларусь, 28 октября 2015. – С. 38-39.
3. Хаджинов М. К. Сервопривод с комбинированным управлением / М. К. Хаджинов, А. Т. Доманов, А. В. Павлова // Информационные технологии и системы 2019 (ИТС 2019): материалы международной научной конференции, БГУИР, Минск, Беларусь, 30 октября 2019. – С. 90-91.

ТЕСТОВАЯ СИСТЕМА ДЛЯ ФУНКЦИОНАЛЬНОЙ ВЕРИФИКАЦИИ ДИСКРЕТНЫХ УСТРОЙСТВ С ПАРАЛЛЕЛИЗМОМ ПОВЕДЕНИЯ

Черемисинова Л. Д., Черемисинов Д. И.

Объединённый институт проблем информатики Национальной академии наук Беларуси
Минск, Республика Беларусь

E-mail: {cld, cher}@newman.bas-net.by

Рассматривается задача, возникающая при моделировании аппаратных реализаций дискретных систем с параллелизмом поведения. Предлагается метод построения тестовой системы для генерации модельных тестов на основе спецификации на проектирование. В основе метода лежит построение TLM моделей описаний на языке параллельных алгоритмов управления.

ВВЕДЕНИЕ

В настоящее время сложность аппаратуры возросла до такого уровня, что требуется развитие новых методов и средств функционального тестирования, которые автоматически генерируют тестовые последовательности и оценивают правильность поведения аппаратной реализации системы. При этом, генерируемые тесты не должны быть жестко привязаны к этой реализации. В настоящее время наиболее развитыми подходами и средствами разработки функциональных тестов являются технологии AVM (Advanced Verification Methodology) [1] компании Mentor Graphics и OVM (Open Verification Methodology) [2] Mentor Graphics и Cadence Design Systems на основе языков SystemVerilog или SystemC. Можно упомянуть также академическую разработку – технологию UniTESK (Unified TEsting and Specification tool Kit) [3] Института системного программирования РАН. В ней основной упор делается на автоматизацию процесса генерации тестовых последовательностей на основе автоматных моделей. Тестовые воздействия находятся в процессе обхода графа состояний конечного автомата, моделирующего тестируемую систему.

В настоящей работе так же, как и в UniTESK, рассматривается задача генерации тестовых последовательностей для функциональной верификации аппаратуры систем управления. Отличие состоит в том, что рассматриваются системы управления с параллелизмом поведения. При разработке тестовой системы используется объектно-ориентированный подход на основе моделирования на уровне транзакций – TLM (Transaction-Level Modeling) [4].

I. ВЕРИФИКАЦИЯ НА ОСНОВЕ TLM МОДЕЛИ

TLM представляет собой подход к моделированию обмена данными в цифровых системах, при котором организация связи между функциональными компонентами отделена от их реализации. На уровне транзакций упор делается на функциональность обмена данными и меньше на их фактическую реализацию. При моде-

лировании на уровне описания аппаратуры, как правило, используется тактовая синхронизация, а на уровне TLM время не используется, синхронизация выполняется, когда данные передаются между процессами. Все выполняемые при этом операции изменения состояний, передача данных и вычисления являются транзакциями. Модель уровня транзакций описывает компонент системы набором процессов, которые определяют его поведение и взаимодействуют одновременно. Каждая транзакция описывает преобразование данных, которые являются общими для взаимодействующих процессов.

Транзакции создаются динамически в процессе моделирования в соответствии с заданными условиями. Произведенные ими изменения состояний данных становятся видимыми для остальных процессов после их завершения.

TLM модели используются в современных средствах верификации проектов аппаратуры. Такой подход позволяет отделить процесс отладки системы и генерации тестовых последовательностей от ее аппаратной реализации.

II. ПОСТАНОВКА ЗАДАЧИ

В настоящей работе принят подход к верификации схемных реализаций систем управления на основе модели (model checking). С целью минимизации размера теста и в то же время максимизации вероятности обнаружения ошибок, ставится задача охвата тестами тех ситуаций, которые возможны в тестируемой системе в процессе ее штатной работы и отражены соответственно в ее спецификации, поэтому тестовые последовательности извлекаются из спецификации системы алгоритмическим путем и применяются к аппаратной реализации системы с тем, чтобы убедиться, что последняя ведет себя корректно, сравнивая ее выходные реакции с теми, которые предусмотрены в спецификации.

Рассматривается случай реактивных систем [5], состоящих из одновременно и независимо работающих компонентов. Важнейшим свойством таких систем является также присущий им параллелизм происходящих в них процессов.

Спецификация на проектирование систем с параллелизмом поведения задается на языке ПРАЛУ описания параллельных алгоритмов управления [5]. Этот язык используется для задания временной упорядоченности событий, возникающих при работе не только самого устройства управления, но и системы в целом, включая его внешнее окружение и описание поведения объекта управления. Реализация устройства рассматривается как черный ящик, для которого доступны только входы и выходы.

Алгоритм управления на ПРАЛУ представляется неупорядоченной совокупностью линейных цепочек l_i операций языка, каждая из которых открывается метками из μ_i и заканчивается метками из ν_i перехода: « $\mu_i : l_i \rightarrow \nu_i$ ». Порядок выполнения цепочек алгоритма управления в процессе его реализации определяется множеством запуска, его текущие значения N_t задают метки тех цепочек, которые могут выполняться одновременно. Если для цепочки « $\mu_i : l_i \rightarrow \nu_i$ » выполняется $\mu_i \subseteq N_t$ и реализуется событие, с ожидания которого начинается цепочка l_i , то она запускается.

III. МОДЕЛИРОВАНИЕ ОПИСАНИЙ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ УПРАВЛЕНИЯ

Традиционно протокол моделируется как набор взаимодействующих процессов, где каждый процесс описывается как конечный автомат. Взаимодействие процессов представляется как коммуникация, в которых актами коммуникации служат транзакции через общие структуры данных, называемые каналами. Для устранения недетерминизма поведения атомарные операции процессов, каковыми являются транзакции, линеаризуются, в результате чего любое параллельное выполнение операций эквивалентно их некоторому последовательному выполнению. Для любого другого потока выполнение линеаризуемой операции является мгновенным: операция либо не начата, либо завершена. Синхронизация процессов модели TLM на уровне транзакций осуществляется барьерным механизмом. Барьер – это точки кода, в которых каждый процесс должен приостановиться и подождать достижения барьера всеми процессами группы. В модели TLM, например, на языке SystemC точки барьера задаются вызовами функции wait.

Ключевые моменты предлагаемой TLM модели описания на языке ПРАЛУ:

1. Произведено уточнение семантики основных операций языка – ожидания и действия. Суть уточнения состоит в доопределении частичного порядка реализации операций, задаваемого параллельным алгоритмом, до линейного. Используется модель параллелизма типа «чередование» (interleaving), в которой одновременность понимается как возможность упорядочивать операции произвольным образом.

2. Операции ожидания и действия рассматриваются в виде композиций линеаризуемых элементарных операций. Транзакции в алгоритмах на ПРАЛУ представлены операциями ожидания и действия, имеющими общую переменную и описывающими событие взаимодействия.

3. Структурой данных в модели TLM алгоритма на ПРАЛУ являются векторы текущих и планируемых значений переменных. Реализация операции ожидания алгоритма на ПРАЛУ состоит в выполнении операций приостановки и проверки текущих значений переменных, операции действия – в выполнении операций установки планируемых значений переменных.

4. Введено понятие ветви как совокупности последовательных процессов, начинающихся с некоторой операции. Ветвь является динамическим объектом, порождаемым операцией образования и уничтожаемым операцией прекращения.

5. Синхронизация параллельных ветвей осуществляется с помощью барьерного механизма. Достижение барьера фиксирует такты работы эмулятора и изменения значений переменных. Точки барьера задаются операцией приостановки выполнения ветвей. Структура данных барьера представлена в памяти очередями ОГ готовых для выполнения ветвей и ОЖ ждущих ветвей.

6. При моделировании алгоритма управления из ОГ последовательно извлекаются ветви и выполняются до приостановки, когда начальный фрагмент ветви G : 1) не может выполняться на множестве текущих значений переменных (тогда ветвь переносится в ОЖ); 2) выполнен (вносится ветвь, начинающаяся с операции, которая должна выполняться в G следующей). При достижении барьера (когда $ОГ = \emptyset$): 1) элементы из ОЖ пересылаются в ОГ; 2) планируемые значения переменных пересылаются в текущие; 3) вводятся новые значения входных переменных.

IV. ЗАКЛЮЧЕНИЕ

Для предлагаемого метода генерации тестов разработана программная поддержка – средства моделирования и синтезаторы ПРАЛУ в модели аппаратуры на языках Verilog и C.

V. СПИСОК ЛИТЕРАТУРЫ

1. Adam Rose, Tom Fitzpatrick, Dave Rich, Harry Foster. Advanced Verification Cookbook. Mentor Graphics Corporation, 2008.
2. OVM Whitepaper. Mentor Graphics Corporation, Cadence, 2007.
3. Кулямин, В.В. Подход UniTesK к разработке тестов / В.В. Кулямин, А.К. Петренко, А.С. Косачев, И.Б. Бурдонов // Программирование. – № 6 (29). – 2003. – С. 25–43.
4. Cai, L. Transaction Level Modeling: An Overview / Lukai Cai, Daniel Gajski // First IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and Systems Synthesis, 1-3 Oct. 2003.– Newport Beach, CA, USA, 2003.
5. Закревский, А.Д. Параллельные алгоритмы логического управления / А.Д. Закревский. – Минск: Ин-т техн. кибернетики НАН Беларуси, 1999. – 202 с.

ПРОГРАММНО-АППАРАТНОЕ СРЕДСТВО «ШКОЛЬНЫЙ ЖУРНАЛ» НА БАЗЕ МИКРОКОМПЬЮТЕРА ORANGE PI ONE С ИСПОЛЬЗОВАНИЕМ БЕСКОНТАКТНЫХ RFID-КАРТ

Шаблинская Е. А., Деменковец Д. В.
Кафедра программного обеспечения информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: ekaterina.vasilevich.01@mail.ru, demenkovets@bsuir.by

В докладе рассмотрены структурная схема программно-аппаратного средства «Школьный журнал», устройство отображения, описание функций программного средства идентификации на основе радиочастотных меток.

ВВЕДЕНИЕ

Одним из принципов научно-технического прогресса является автоматизация. Все чаще организации принимают решение вести отчетность в электронных носителях, а не бумажных.

Бумажный школьный журнал сильно проигрывает электронному по ряду многих факторов: его можно уничтожить физически и данные в нем будут бесследно потеряны, нет гарантии, что данные в нем 100% подлинны и внесены конкретным лицом и тд.

Сегодня RFID-идентификация на основе карт с радиочастотными метками — это достаточно бюджетный и надежный способ обеспечить систему контроля посещаемости и успеваемости учеников. В связи с этим была поставлена цель разработать программно-аппаратный комплекс «Школьный журнал» с разработкой программного обеспечения устройства отображения и программного средства для идентификации учеников на основе персональных карт. В данном докладе рассмотрена организация разработанной системы.

I. ОРГАНИЗАЦИЯ ПРОГРАММНО-АППАРАТНОГО СРЕДСТВА

Была разработан программно-аппаратный комплекс, состоящий из следующих компонентов:

- устройства идентификации;
- устройства отображения

В свою очередь устройство идентификации состоит из следующих компонентов:

- микрокомпьютер Orange Pi One;
- считыватель радиочастотных меток;
- модуль Ethernet;
- блок питания.

Устройство отображения представлено в двух видах:

- веб-приложение для ПК;
- мобильное приложения для устройства на базе Android.

Также используются персональные карты с радиочастотными метками.

Структурная схема системы отображения изображена на рисунке 1.

Взаимодействие мобильного устройства отображения с устройством идентификации выполняется посредством связи Internet используя REST API.

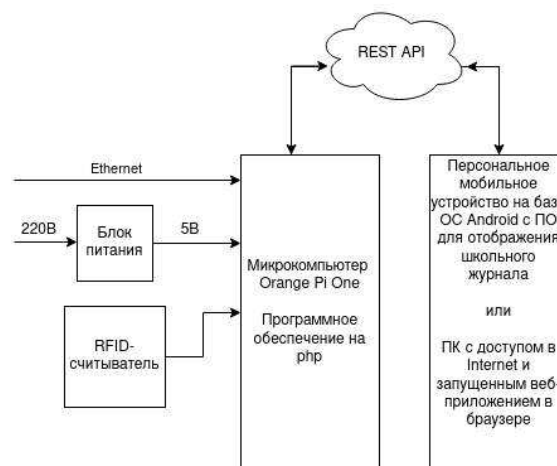


Рис. 1 – Структурная схема программно-аппаратного средства «Школьный журнал»

Основные проблемные вопросы возникающие при создании подобных комплексов являются:

- организация постоянной и корректной связи между устройством идентификации и устройством отображения;
- правильная организация базы данных;
- организация доступа к устройству отображения в соответствии с правами;
- степень защиты от окружающей среды и антивандальность;
- возможность конфигурации с мобильного устройства (смартфона) без использования проводного соединения;
- пользовательский интерфейс на русском языке;
- энергопотребление;
- охлаждение;
- стоимость.

Основные проблемные вопросы удалось решить в разработанном протоколе взаимодействия между устройством отображения и устройством идентификации.

Обмен данными между двумя устройствами будет происходить с использованием сервера для хранения базы данных и самого веб-приложения.

Степень защиты устройства отображения предложено решить с помощью корпуса, который выполнен из пластика.

Т.к. микрокомпьютеры имеют склонность к перегреву, было решено инициировать работу устройства только на некоторый малый промежуток времени по определенному графику работы (раз в 10 минут с 7:30 до 21:00), это решит проблемы с нагревом устройства, а, соответственно, и охлаждением.

Организация прав доступа к устройству осуществляется с помощью паролей и уровней доступа.

II. ПРОГРАММНОЕ СРЕДСТВО КОМПЛЕКСА И УСТРОЙСТВО ОТОБРАЖЕНИЯ

Программное средство комплекса позволяет производить идентификацию и управление устройством отображения в соответствии с правами доступа.

Технические характеристики и возможности программного средства:

- поддержка уровней доступа к устройству отображения с мобильного устройства (родитель/ученик, учитель, директор);
- авторизация с помощью пароля;
- управление с мобильного устройства;
- передача данных с устройства идентификации на сервер посредством Internet и REST API;
- получение данных с сервера пользовательским приложением на основе уровней доступа посредством REST API;
- сохранение конфигурации в энергонезависимой памяти;
- отображение текущих идентифицированных пользователей;
- возможность коммуникации между пользователями;
- степень защиты устройства отображения IP65;
- антивандальный защитный корпус.

Разработка программы в микрокомпьютере выполнена на языке PHP с использованием Bash Scripting. В качестве языка программирования для мобильного программного средства по отображению информации был выбран язык Kotlin. Данный язык был представлен как официальный язык разработки для платформы Android 17 мая 2017 года [3,4]. Для веб-приложения средства отображения был выбран стек PHP/Аpache. В качестве СУБД используется MySQL.

На включенном и подключенном к сети микрокомпьютере, с установленной ОС Ubuntu 20.04, программа-демон раз в 10 минут отправляет на сервер файл, где указан Id учителя, время отправки, id всех отметившихся учеников, номер кабинета, в котором установлен считыватель. На сервере эта информация записывается в базу данных.

Функциональность, доступная в аккаунте учителя:

- возможность записать план занятий по предмету;
- отображение ФИО всех учеников в классе;
- отображение отсутствующих учеников в классе;
- возможность оценить ученика;
- возможность оставить заметку для родителей;
- возможность записать домашнее задание.

Функциональность, доступная в аккаунте ученика/родителя:

- отображение посещаемости;
- отображение табеля успеваемости;
- отображение домашнего задания;
- отображение замечаний учителя.

Функциональность, доступная в аккаунте директора:

- отображение таблицы успеваемости и посещаемости по каждому предмету для каждого класса;
- отображение личных карточек всех пользователей;
- возможность добавить, удалить, редактировать аккаунт пользователя;
- возможность оставлять замечания для пользователей.

III. ЗАКЛЮЧЕНИЕ

Данное программно-аппаратное средство может иметь широкое применение, быстро и дешево расширяться в своих функциональных возможностях. Программное средство может работать в рамках не только одной школы, но и школ всего города, области и тд.

IV. СПИСОК ЛИТЕРАТУРЫ

1. Orange Pi One Documentation [Электронный ресурс] / - Режим доступа: <http://www.orangepi.org/Docs/mainpage.html>. - Дата доступа: 16.10.2021.
2. RFID Reader USB Port EM4100 [Электронный ресурс] / - Режим доступа: <https://www.newegg.com/p/1W9-00KN-00TG8>. - Дата доступа: 16.10.2021.
3. Android Announces Support for Kotlin [Электронный ресурс] / - Режим доступа: <https://android-developers.googleblog.com/2017/05/android-announces-support-for-kotlin.html>. - Дата доступа: 16.10.2021.
4. Kotlin Documentation [Электронный ресурс] / - Режим доступа: <https://kotlinlang.org/docs/reference/>. - Дата доступа: 16.10.2021.

ОСОБЕННОСТИ РЕАЛИЗАЦИИ ЗАДАЧ ЦИФРОВИЗАЦИИ НА БЕЛОРУССКИХ ПРЕДПРИЯТИЯХ

Шведова О. А.

Кафедра информационных систем и технологий, Институт информационных технологий, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: 6749764@gmail.com

На примере молочного предприятия рассмотрены существующие особенности реализации автоматизации технологических процессов в Беларуси. Проанализированы задачи, которые необходимо решить для проведения успешной модернизации производств для перехода к Индустрии 4.0

ВВЕДЕНИЕ

На примере молочных предприятий Беларуси, которые являются лидерами по производству и поставке готовой продукции за границу, рассмотрим вопросы Индустрии 4.0 в нашей стране. Это одно из лидирующее направлений производства, куда постоянно инвестируются деньги на развитие и модернизацию в зависимости от требований рынков. В настоящее время основным направлением поставок является РФ, которая периодически предъявляет все большее количество различных требований к производителям по осуществлению контроля за исходными продуктами, изготовлением продукции, упаковкой, логистикой и реализацией.

I. Производство молочной продукции

Технологический процесс производства молочной продукции состоит из множества этапов, которые выполняются в определенной последовательности различными установками и линиями, которые могут работать на предприятии как отдельные автоматизированные единицы, а могут входить в состав более сложных систем управления технологическими процессами, производствами, предприятиями, соответственно SCADA-систем, MES систем или ERP систем.

Например, последовательность обработки молока может выглядеть следующим образом:

1. получение молока – доильные комплексы;
2. первоначальная очистка и хранение молока – сепараторы и резервуары для охлаждения;
3. перевозка молока – молоковозы;
4. приемка молока на предприятии – посты приемки молока;
5. хранение молока – распределительная гребенка и резервуары хранения;
6. очищение молока – сепараторы;
7. первоначальная обработка – пастеризация;
8. переработка:
 - изготовление цельномолочной продукции – линии изготовления молока, сметаны, масла и др.;
 - изготовление кисломолочной продукции – линии сквашивания;

- изготовление сыров – сыроизготовительные линии;
 - изготовление сухой продукции – сушильные установки;
9. упаковка;
 10. хранение;
 11. поставка.

Для соблюдения требований рынка часто возникает необходимость адаптировать существующие установки. Например, сейчас актуальной является модернизация упаковочных машин и автоматов для соответствия требованиям системы «Честный знак» - система цифровой маркировки с использованием кодов DataMatrix, с чем связано выявление большого количества нюансов в реализации данной задачи: нанесение этикеток, контроль использования этикеток, контроль брака продукции, действия с испорченными этикетками и т.д.

Проведение данных мероприятий обострило существующие вопросы в производственных технологических цепочках, ещё раз напомнив нам, что текущее состояние производств с точки зрения индустриализации относится больше к Индустрии 3.0, нежели к 4.0, которая началась в 2013 году.

II. Цифровизация производства

Разновидностью IoT является промышленный (индустриальный) интернет вещей (Industrial Internet of Things, IIoT). Именно он открывает прямую дорогу к созданию полностью автоматизированных производств.[1]

В начале внедрения IIoT на промышленное оборудование устанавливают датчики, исполнительные механизмы, контроллеры и человеко-машинные интерфейсы. В результате становится возможным сбор информации, которая позволяет руководству получать объективные и точные данные о состоянии производства. Обработанные данные предоставляются всем подразделениям предприятия. Это помогает наладить взаимодействие между сотрудниками разных подразделений и принимать обоснованные решения. При обработке огромного массива неструктурированных данных, поступающих с датчиков, их фильтрация и адекватная интерпретация становится

приоритетной задачей. Поэтому особую значимость приобретает представление информации в понятном пользователю виде. Для этого используются передовые аналитические платформы, предназначенные для сбора, хранения и анализа данных о технологических процессах и событиях, работающие в реальном масштабе времени.

Задача-максимум состоит в достижении такого уровня автоматизации предприятия, при котором на всех участках, где это возможно, машины работают без участия людей. Роль персонала при этом сводится к контролю работы машин и реагированию лишь на экстренные ситуации.

По мере становления цифровых экосистем производственные предприятия из изолированных систем, самостоятельно выполняющих все необходимые для производства продукции производственные и бизнес-процессы, будут преобразовываться в открытые системы, объединяющие различных участников рынка; управлять средствами производства в этих системах будет не персонал, а облачные сервисы, конечная цель всех этих трансформаций – не выпуск продукции, а предоставление услуг потребителю.

Главными составляющими IoT становятся передовые аналитические инструменты, искусственный интеллект и машинное обучение.

На сегодняшний день мире есть три гранда, которые выпускают программы для автоматизации промышленности: Siemens PLM, PTC Pro/ENGINEER, IBM PLM CATIA; а так же небольшие инжиниринговые компании в разных странах мира по различным направлениям предлагают готовые программные решения для осуществления перехода на новый этап автоматизации - цифровизации производств.[2]

III. МОДЕЛИРОВАНИЕ ДЛЯ МОДЕРНИЗАЦИИ ПРОИЗВОДСТВА

Моделирование производственных технологических процессов является одной из наиболее важных составляющих, что позволяет анализировать текущее состояние предприятия, наиболее рационально решать задачи модернизации существующих автоматизированных систем управления, разрабатывать и планировать пере-

оснащение, увеличение производственных мощностей. Так же это даёт возможность достойно конкурировать на местном и зарубежных рынках по поставкам товаров, сократить издержки, обеспечить экономию природных ресурсов, наиболее гибко адаптировать продукцию к актуальным требованиям заказчиков.

В настоящее время далеко не все промышленные предприятия РБ прошли этап Индустрии 3.0 - Автоматизация технологических процессов с возможностью организации мониторинга, управления и сбора данных для различных установок. Без успешного завершения данного этапа невозможно реализовать переход на следующий, т.к. он основывается на существующей базе.

Для создания оптимальной основы для внедрения технологий Индустрии 4.0 (IoT) необходимо очень осознанно подходить к решению задач Индустрии 3.0 для обеспечения оптимального эффекта, более быстрого и экономически целесообразного дальнейшего перехода. Конкурировать на рынке продукции без технологий Индустрии 4.0 в ближайшем будущем уже не представляется возможным из-за колоссальных преимуществ нововведений.

ЗАКЛЮЧЕНИЕ

Основная задача исследований и разработки - по результатам анализа текущего уровня автоматизации технологических процессов предприятия, предлагаемого программно-аппаратного комплекса Сименс, разработать модели и алгоритмы проведения поэтапной модернизации предприятий для перехода на этап Индустрия 4.0 и его реализации.

Таким образом вопросы планирования и моделирования становятся очень актуальными для промышленных предприятий РБ, которые нацелены на успешное развитие бизнеса в условиях существующей конкуренции.

IV. СПИСОК ЛИТЕРАТУРЫ

1. Индустрия 4.0 [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://www.it.ua/ru/knowledge-base/technology-innovation/industry-4>
2. Индустрия 4.0 в Беларуси: быть или не быть? [Электронный ресурс]. – Электронные данные. – Режим доступа:

СИСТЕМА УПРАВЛЕНИЯ ПОДДЕРЖАНИЯ ТРАЕКТОРИИ ПОЛЕТА БПЛА ПРИ МОНИТОРИНГЕ ОКРУЖАЮЩЕЙ СРЕДЫ С ИСПОЛЬЗОВАНИЕМ ТЕОРИИ НЕЧЁТКИХ МНОЖЕСТВ

Шумский А. Н., Карпович Д. С.

Кафедра автоматизации производственных процессов и электротехники,

Белорусский государственный технологический университет

Минск, Республика Беларусь

E-mail: andreishumski91@gmail.com, karpovich@tut.by

Представлен способ управления беспилотным летательным аппаратом самолетного типа с применением теории нечетких множеств позволяющий производить мониторинг окружающей среды при полете по маршруту, как вблизи, так и вдали от производства, со стабилизацией бокового отклонения от линии заданного пути.

ВВЕДЕНИЕ

На данный момент, большая часть задач, возлагаемая на беспилотные летательные аппараты (БПЛА), выполняется в автоматизированном режиме. Для управления БПЛА используется автопилот, в котором находится логика его работы, законы управления, а также навигационные решения для полета, при этом человек-оператор контролирует корректное выполнение задач, возложенных на БПЛА. При необходимости человек-оператор может передать команды на борт БПЛА, которые в свою очередь могут корректировать высоту маршрута, скорость полета, полет к точке интереса и др. БПЛА в процессе выполнения миссии опрашивает наземную станцию управления, где находится человек-оператор, данные о высоте, скорости, углах и др. (телеметрию), которые используются для оценивания корректности и соответствия заданным параметрам миссии [1].

I. ПОЛЕТ БПЛА ПО МАРШРУТУ

В настоящее время, ввиду роста вредных производств, а также повышения количества выбрасываемых веществ в атмосферу, появляется необходимость в контроле и регистрации показаний воздушной среды как вблизи, так и вдали от производства, для принятия мер по снижению количества выбросов. Поскольку для получения показаний о состоянии воздуха, необходимо покрывать большую площадь мониторинга, то применение беспилотных летательных аппаратов позволит как ускорить снятие показаний, так и получать данные показания по заданному интервалу времени, а также по строго заданному маршруту, что в свою очередь даёт объективно оценивать величину выбросов. Для полета беспилотного летательного аппарата необходимо создание маршрута, в соответствии с которым, БПЛА будет выполнять свои задачи. При этом следует учитывать, что между поворотными пунктами маршрута (ППМ) можно лететь по двум из следующих способов [2,3]:

- курсовой способ наведения, заключается в выдерживании заданного курса от текущего местоположения на следующий ППМ;
- маршрутный способ наведения, заключается в построении прямой линии заданного пути (ЛЗП) между двумя ППМ и последующим полётом с выдерживанием ЛА на ней.

Целью системы автоматического управления БПЛА на этапе полета по заданному маршруту является выдерживание заданной траектории горизонтального полета [4]. Поскольку выдерживание ЛЗП должно осуществляться с максимальной точностью, то необходимо использовать закон управления боковой координатой с интегральной составляющей (1).

$$\delta_{aileron} = K_Z \dot{Z} + K_Z Z + K_Z^f \int Z dt.$$

Так как минимизация отклонения от ЛЗП происходит с помощью элеронов, то в закон управления необходимо добавить составляющие, которые будут ограничивать максимальный крен при прохождении ППМ.

$$\delta_{aileron} = K_{\omega_x} \omega_x + K_{\gamma} \gamma + K_Z \dot{Z} + K_Z Z + K_Z^f \int Z dt.$$

Структурная схема системы автоматического управления БПЛА при полете по маршруту и стабилизации бокового отклонения от ЛЗП представлена на рис. 1.

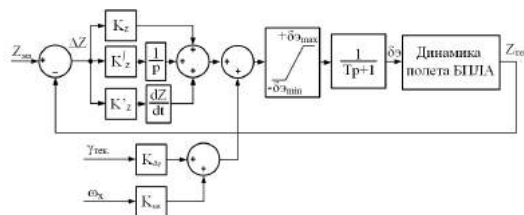


Рис. 1 – Структура системы управления

Параметры закона управления находились с использованием нечеткого регулятора [5,6]. Тогда структурная схема контура управления примет вид (см. рис. 2).

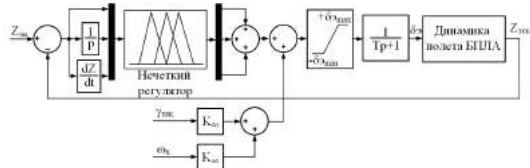


Рис. 2 – Структура системы управления с нечетким регулятором

На рис. 2 представлены нечеткий ПИД-регулятор для бокового отклонения от ЛЗП. ПД-регулятор защиты от максимального крена. Упрощённая модель исполнительного механизма (рулевая машинка), а также модуль динамики полета БПЛА.

II. РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЙ

В режиме полета по маршруту элероны минимизируют отклонение от ЛЗП между двумя ППМ. При прохождении одного из ППМ происходит смена курса (курс на следующий ППМ), поэтому после прохождения ППМ наблюдается $\frac{3}{4}$ скачок; на графике бокового отклонения, которое компенсируется работой элеронов и руля направления. График переходного процесса бокового отклонения от ЛЗП (см. рис. 3).

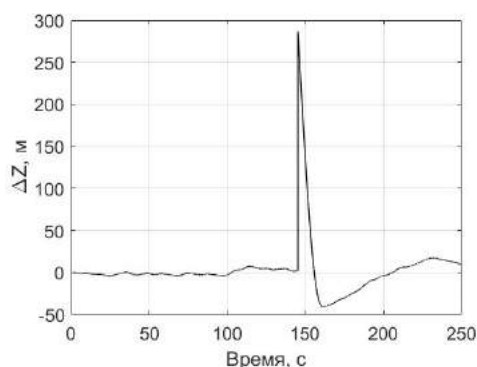


Рис. 3 – Переходной процесс бокового отклонения от ЛЗП

График переходного процесса крена (см. рис. 4).

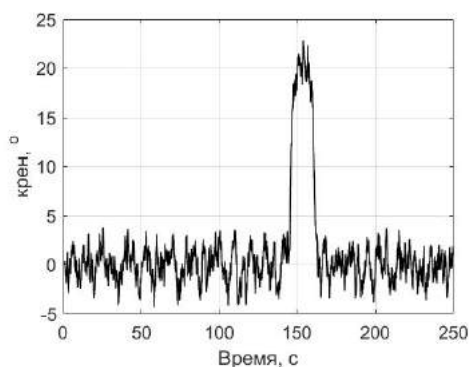


Рис. 4 – Переходной процесс изменения крена

График переходного процесса отработки элеронов (см. рис. 5).

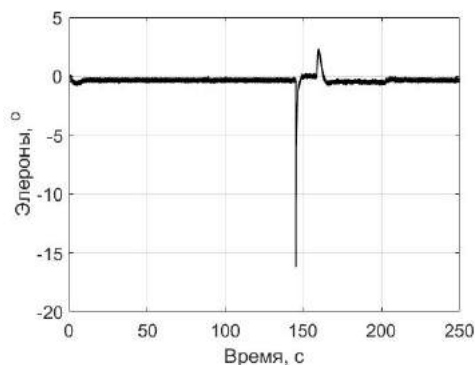


Рис. 5 – Переходной процесс работы элеронов при полете по маршруту

III. ЗАКЛЮЧЕНИЕ

В соответствии с полученными результатами для системы автоматического управления БПЛА при полете по заданному маршруту, можно отметить, что закон управления боковым отклонением от ЛЗП, синтезированный с помощью теории нечетких множеств, может быть использован в САУ для мониторинга окружающей среды, где важна точность соблюдения полета по маршруту, а также необходима повторяемость полета БПЛА по заданному пути неограниченное количество раз. Данная повторяемость позволит набрать определенное количество измерений в заданных точках отбора проб, для составления полной картины статистики выбросов нежелательных веществ в окружающую среду. Что в свою очередь, позволит принять необходимые меры по снижению неблагоприятных выбросов в атмосферу.

IV. СПИСОК ЛИТЕРАТУРЫ

1. Способы организации управления беспилотными летательными аппаратами / Иванова И. А., Никонов В. В., Царева А. А. // Актуальные проблемы гуманитарных и естественных наук, -- 2014. -- No 11-1. -- С. 56–63.
2. Распопов, В. Я. Микросистемная авионика: учебное пособие. -- Тула : Эгриф К 2010. – 248 с.
3. Романова, И. К. Методы синтеза системы управления летательными аппаратами: учебное пособие. -- Москва: Издательство МГТУ им. Н. Э. Баумана, – 2017. -- 153 с.
4. H. Chao, Y. Luo, L. Di, and Y. Q. Chen. Fractional order flight control of a small fixed-wing uav: controller design and simulation study / Proc. ASME Int. Design Engineering Technical Conf. Computers and Information in Engineering, No. MESA-87574, – 2009.
5. Jin J., Huang H., Sun J., Pang Y. Study on Fuzzy SelfAdaptive PID Control System of Biomass Boiler Drum Water / Journal of Sustainable Bioenergy Systems. -- 2013. -- V. 3. -- P. 93–98.
6. Демидова Л. А. Принятие решений в условиях неопределенности / Демидова Л. А., Кираковский В. В., Пылькин А. Н. – М.: Горячая линия -- Телеком, – 2012. – 290 с.

ANALYSIS OF ARTIFICIAL NEURAL NETWORK IN A PHOTOVOLTAIC SYSTEM TO EXTRACT THE MAXIMUM POWER IN A PHOTOVOLTAIC BASED SYSTEM PARADIGM

Elzein Imad

Department of Electrical Engineering, Belarusian National Technical University

Minsk, Belarus

E-mail: elzein1@hotmail.com

When implementing a photovoltaic system, different parameters play an integral part in enhancing the extraction of the maximum power of a solar setup. One of those parameters is the maximum power point tracking (MPPT) algorithm which is used to track and extract the maximum power P_{max} when we refer to PV energy productivity. In this research, we will introduce a new scheme of tracking the P_{max} using an artificial neural network (ANN) in different dynamic environmental conditions. The proposed artificial neural network (ANN) based PV model is going to be applied and utilized with a perturb and observe (P&O) algorithm. The improvement of using ANN in PV system would improve its ability to detect MPP at a short period of time as compared to other available methods.

INTRODUCTION

As we start thinking of renewable energy and photovoltaic systems the theme of maximum power point tracking algorithm is currently moving forward to play such a vital and essential role as we start thinking of the efficiency of the solar panel. The future of solar panels is promising according to many literature surveys and the commencement of thinking to rely on solar energy encouraged us to develop an approach to benefit from these solar radiations and convert them into a new transformation in the form of electrical energy. In fact we have to admit that the high demand and reliance on electricity worldwide, and since this source cannot keep abreast with appropriate and sufficient delivery of this supply, high electricity bills, uprising fuel prices, and pollution drove us to integrate and implement the usage of photovoltaic energy to be another alternative and promising source of energy as it is, accessible, environmentally very promising as we stress out clean environment, with an extra bonus of decreased bill to operate. In the past years it was proved that photovoltaic systems were merged gradually in standalone and grid connected modes of PV systems. [1]

It is known by so far that a PV system arrays is used to convert sunlight into electricity. The quantity of the direct current generated takes into consideration two parameters; where it relies on the irradiance and temperature. In addition to that a variation may take place due to the other parameter in a PV system which is the load.

As known from a (P-V) curve of a solar panel, there is an optimum operating point such that the PV delivers the maximum possible power to the load. The optimum operating point changes with the solar irradiation, and cell temperature. Therefore, on line tracking of the maximum power point of a PV array is an essential part of any successful PV system. A variety of maximum power

point tracking (MPPT) methods are developed. The methods vary in implementation complexity, sensed parameters, and required number of sensors, convergence speed, and cost [2].

When a PV system operates at the standard test conditions and through a uniform temperature and irradiance, the photovoltaic arrays exhibits an I-V characteristic along with a maximum power point. This P_{max} point is important to be tracked to allow the PV array generates a maximum output power and this is can be performed by adapting a maximum power point tracking (MPPT). Many researchers investigated a range of MPPT algorithms that are well known and each has its own advantages and disadvantages [3].

Comparison among those MPPT leads to investigate parameters such as convergence speed and the work done was related to the time of detecting the maximum power point MPP [4].

I. LITERATURE REVIEW

One of the controllers applied in PV systems was fuzzy logic which was applied in manipulating various MPPT controllers. The main theme of fuzzy logic relies on constructing linguistic rules sets to conclude the required change in duty cycle (ΔD), and its inputs are an error (E) in the current or voltage of PV panel and the change in error (CE).

The new approach of integrating neural networks and its applications in estimating maximum power generation was addressed as per the voltage corresponding to the maximum power based on reference open circuit voltage of the PV panel and a time parameter. In other literature reviews the maximum power generation was calculated based on two environmental conditions known through the irradiation and temperature levels.

II. PV SYSTEM

The PV system considered in this paper is shown in Fig. 1. The following is discretion of each component. The system consists of a PV solar panel, a dc-dc power converter, controller and a system load.

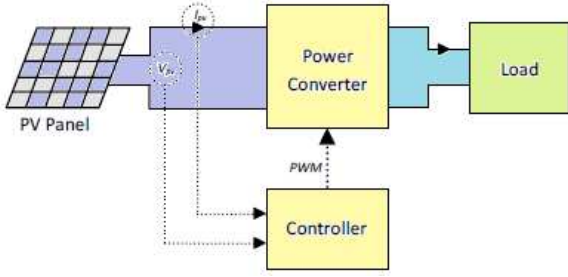


FIG. 1 – PV system architecture

As a result of the majority of utilized methods implemented to seek MPPT, experts in the field of PV systems have displayed an overview of investigating maximum power point tracking procedures. Indeed, few papers present relative research and studies among a narrow set of methods. To select the suitable and desirable maximum power point, proper tracking will be presented between the PV system and definitely the load. Efficiency plays an integral role in PV systems and the photovoltaic is influenced by irradiance falling onto the solar panel and temperature which cannot be forecasted or being known ahead of time, hence, it's unrealistic if we are seeking maximum power to go ahead and make a direct interconnectivity between the load and the PV and thus to create what is known as a photovoltaic balance of system (BOS). Technically speaking balance of system is active inside DC-DC converters and its role is to alter the characteristics of the load. Thus, such converter has a function of maximizing the power as being sent to the load. Hence, the important role of a DC/DC converter in a PV system is primarily being accountable for transferring P_{max} from the solar panel to the load. This goes about as conformity to match impedance of source and load. Such converters are equipped with MPPT and those DC-DC converters can produce a low or high DC voltage. Maximum power point (MPP) position is unfortunately unknown; however, it can be located by calculation models or via search algorithms. [5]

III. ARTIFICIAL NEURAL NETWORKS

In this research the approach is to apply an analysis study of a neural network to provide thorough accuracy in reaching the maximum power point (MPP) through the utilization of an MPPT tracking algorithm such as the Perturb and Observe (P&O) algorithm.

Identifying the maximum power point (MPP) will provide the proper feedback for the controller to

take the appropriate steps in moving the operating point towards the maximum power point through a set of accomplishments, namely, a faster response time and steady reliable tracking of the optimal power point. Fig. 2 shows the PV system with the neural network and P&O MPPT controller.

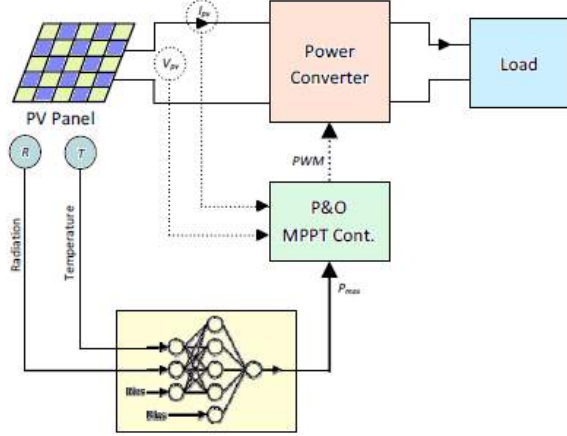


FIG. 2 – PV system module

It is important to note that the objective of the designed neural network is not to track the MPP, but to help the controller achieve a better tracking performance by providing it with the estimated maximum power corresponding to the current atmospheric conditions. In the previous section, fuzzy logic was applied in the controller itself as a tracking strategy.

The P&O MPPT strategy is simple and easy to implement. However, it experiences a trade-off between tracking time and output power stability. Knowing the value of the maximum generated power (P_{max}) can help in overcoming this trade-off problem and enhance the performance of the controller.

The design of the neural network and the controller are now discussed.

IV. MPPT CONTROLLER BASED ON NEURAL NETWORK

Over the past years many research studies have been addressing ANN based techniques. In this regard, the main parameters of a photovoltaic system module are the well known two variables; temperature (T) and irradiance (G). These variables are fed to an ANN as incoming variables.

At the other output stage, the output is represented through the duty ratio acquired and then forwarded to a DC-DC boost converter that regulate and control the voltage to its optimal value.

Figure 3 demonstrates the proposed MPPT setup as shown below.

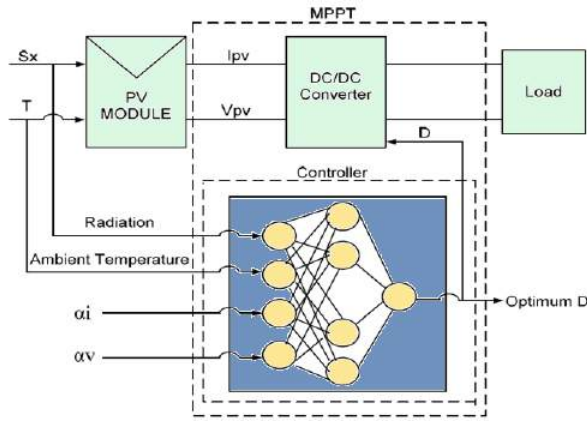


Рис. 3 – ANN proposed MPPT module

An input value is assigned at the incoming layer and so forth for the next layers where hidden layer encompasses several neurons received after the 1st layer and then forward the values to the output layer. After this process is accomplished the output layer delivers values of the duty ratio to the main module.

In the following equations the input and output values are carried out:

$$y_j^h = f\left(\sum_{i=1}^N W_{ij} X_i + \theta_j^h\right) \quad (1)$$

$$y_k^o = f\left(\sum_{j=1}^{Nh} W_{kj} Y_j^h + \theta_k^o\right) \quad (2)$$

In this analysis we propose two phases and stages to track the maximum power point MPP.

During the first phase a set of trained data (ANN) is used to drive the reference current (I_{ref}) and reference voltage (V_{ref}) to an optimal current (I_{opt}) and optimal voltage (V_{opt}) to reach the maximum power point.

V. CONCLUSION

This paper addressed an analysis of applying an artificial neural network in a photovoltaic system to extract the maximum power in a photovoltaic based system. The discussion throughout this analysis was to suggest utilizing the ANN due to its fast response to changes with respect to dynamic weather factors and its accuracy in detecting the maximum power point.

1. I. Elzein. Integration of a Predictive Control Model and MPPT for PV Station. International Conference on Smart Systems and Technologies (SST). IEEE proc., ISBN: 978-1-5090-3720-9, Oct 2016.
2. M. Elzein. Integrating a P&O through a Variable Steps Fuzzy Logic Control for a Photovoltaic System to Track the Maximum Power Point. International Conference on Advances in Computing and Technology (ICACT-2020) Proceedings.
3. Trishan Esham, Patrick L. Chapman, Comparison of Photovoltaic Array Maximum Power Point Tracking Techniques,"IEEE Trans. Energy Conversion, vol 22, No.2 June 2007.
4. Carlos A.P.Tavares,Karla T.F.Leite, Walter I. Suemitsu, Maria D.Bellar "Performance evaluation of photovoltaic Solar system with different MPPT methods,"in IECON 2009 proceedings, Page 716-721.
5. Elzein, I. Maximum Power Point Tracking System for Photovoltaic Station: A Review. Journal of System Analysis and Applied Information Science, Minsk, Vol.3, 2015, pp.15-20. Universal Decimal Classification (UDC) 620.92+502.174.3

Секция «Интеллектуальные
информационные технологии»

Section «Intelligent information
technologies»

РАЗРАБОТКА АЛГОРИТМА ДОБАВЛЕНИЯ ИСТОЧНИКОВ СВЕТА В КОМПЬЮТЕРНЫЕ СЦЕНЫ

Адамюк А. А., Щегрикович Д. В.

Факультет радиофизики и компьютерных технологий, кафедра интеллектуальных систем,

Белорусский государственный университет

Минск, Республика Беларусь

E-mail: rct.adamyuk@bsu.by

В статье описана идея алгоритма добавления искусственного источника света в компьютерные сцены. Разработан план архитектуры с применением математической модели нейронных сетей. Объясняется принцип построения синтезированного датасета.

ВВЕДЕНИЕ

Целью работы является создание алгоритма внесения искусственного источника света в цифровое изображение. Предполагалась возможность применения алгоритма в сфере разработки игр 2D-графики. Принимая как входной параметр информацию о вносимом источнике света, алгоритм модифицирует изображение в соответствии с полученной информацией и контекстом картины.

Родственными для данной задачи являются такие задачи компьютерной графики, как шейдинг, рендеринг. Тема оценки освещения часто поднимается в области дополненной реальности.

I. НАПРАВЛЕНИЕ АЛГОРИТМА

Уравнение рендеринга – модель, основанная на входящей в камеру энергетической яркости (англ. radiance), принимающая на вход данные о позиции, нормалях и отражающей способности материалов на сцене. Эта модель, реализующаяся такими методами, как фотонные карты, трассировка путей Монте-Карло и методом конечных элементов, применяемом в методе излучательности (англ. radiosity), является достаточно вычислительно затратной. [1]

Среди математических моделей особую популярность получила такая модель, как нейронная сеть. Применение ее в алгоритме отрисовывания света должно привести к уменьшению вычислительной стоимости задачи.

Также можно использовать предположение о том, что сцены состоят из небольших плоских поверхностей, а глубина всех пикселей, принадлежащих поверхности, может быть вычислена по трехмерному местоположению и ориентации поверхности, которой они принадлежат. Это в основном означает, что даже самые сложные трехмерные сцены могут быть выражены с помощью трехмерного местоположения и ориентации небольших поверхностей. Справедливость этого предположения можно увидеть в графических движках, где многие сложные модели могут быть созданы с простыми треугольными поверхностями. [2]

Предполагая, что алгоритм может быть использован для облегчения работы 2D-художника при работе со сценами с динамическим светом, было решено создать алгоритм, получающий одно изображение на вход. Чтобы обеспечить возможность изменения положения объектов на сцене в реальном времени, также было решено отказаться от методов, использующих предварительное вычисление схемы распространения света на конкретной сцене, таких, как предварительно вычисленная передача излучения (англ. PRT). Также, выбирая направление алгоритма, отдавалось предпочтение не использующим восстановление 3D-модели, так как обработка 3D достаточно ресурсоемкая задача.

Разрабатываемый алгоритм можно охарактеризовать как алгоритм одного изображения (англ. single image), учитывающий объекты лишь области экрана (англ. screen space).

II. ОЦЕНКА ГЕОМЕТРИИ СЦЕНЫ

Чтобы лучше понимать оцениваемое окружение, исследователи подключают дополнительные модули, возвращающие дополнительные сведения о сцене. В контексте нейросетей подобное можно осуществить несколькими способами. Например, связать определенные слои нейросети основной задачи с вспомогательной или же применять функцию потерь, представляющую собой суммарную потерю от обеих задач.

Задача не является основной, но, так как ее ошибки напрямую влияют на результат основной задачи, нам необходимо серьезно изучить научную область дополнительной задачи.

В данной работе мы извлекаем данные о геометрии сцены из входного изображения с помощью дополнительных свёрточных нейросетей. Полученная информация о глубине и нормалях сцены передается дальше в свёрточную нейросеть, ответственную за отрисовывание света.

Оценка глубины по одному изображению достаточно сложная задача. Для внесения большей определенности в алгоритмах оценки глубины одного изображения используются следующие подсказки:

- Положение на изображении. Объекты, которые находятся дальше, обычно ближе к горизонту.
- Окклюзия. Объекты, которые расположены ближе, закрывают те, что находятся позади них. Окклюзия предоставляет информацию о порядке глубины, но не о расстоянии.
- Плотность текстуры. Текстурированные поверхности, расположенные дальше, выглядят на изображении более мелкозернистыми.
- Линейная перспектива. Прямые параллельные линии физического мира кажутся сходящимися на изображении.
- Видимый размер объектов. Объекты, которые находятся дальше, кажутся меньше.
- Затенение и освещение. Поверхности кажутся ярче, когда их нормаль направлена на источник света. Часто предполагается, что свет исходит сверху. Затенение обычно предоставляет информацию об изменениях глубины на поверхности, а не относительно других частей изображения.
- Размытие фокуса. Объекты, которые находятся впереди или позади фокальной плоскости, выглядят размытыми.
- Воздушная перспектива. Очень далекие объекты (километры) менее контрастны и приобретают голубоватый оттенок.[3]

Для этой задачи оценки глубины было решено использовать обученную нейросеть алгоритма 3D Ken Burns Effect from a Single Image [4], используя технику переноса обучения. Архитектура их нейросети имеет 3 этапа оценки глубины: с использованными архитектурами GridNet, Mask R-CNN и U-Net, – с сильным упором на семантическую сегментацию и сегментацию экземпляров.

III. СИНТЕЗИРОВАНИЕ ДАТАСЕТА

В настоящее время большой проблемой в сфере оценки освещенности является разобщенность условий, при которых создается датасет, и проблема качества, такая как низкий динамический диапазон, порождающая артефакты засветленности на ярких участках картины, приводящие к неразличимости деталей на этих участках.

Было принято решение синтезировать собственный датасет, соответствующий нашей задаче. Будет использоваться движок Unity для создания снимков сцен с разных ракурсов. Для каждого ракурса сцены будут отсняты карты глубины и нормалей, несколько изображений с разным положением точечного источника света и одно с рассеянным глобальным светом. Композиция объектов на сценах датасета разной слож-

ности, что обеспечивает возможность поэтапного обучения нейросети. Сначала обучение проводится на основных геометрических телах, нейросеть учится базовым признакам, потом показываются более сложные объекты.

IV. АРХИТЕКТУРА НЕЙРОННОЙ СЕТИ

Предполагается, что на входном изображении присутствует рассеянное освещение, то есть в картине нет точечных источников света, свет не направлен, а вносимый источник света ламбертовский.

Для нейросети, отвечающей за отрисовку источника света, было решено взять архитектуру условного вариационного автоэнкодера (англ. cVAE). После обучения на датасете вариационный автоэнкодер хранит распределение скрытых значений, характеризующих датасет. Далее автоэнкодер можно применять для генерирования изображений в соответствии с известным ему распределением. Нейросеть решает задачу мультимодального переноса изображения в изображение (англ. Multimodal I2IT). I2IT – задача преобразования входного изображения в целевое изображение. Мультимодальное I2IT предполагает несколько изображений на выходе.

Получая на вход изображение, нейросеть должна генерировать изображения в соответствии с указанным положением источника света.

V. ЗАКЛЮЧЕНИЕ

Была разработана архитектура алгоритма добавления источника света в изображение. Желание разработать алгоритм, удобный в применении обычными пользователями, привело к некоторым ограничениям в возможностях алгоритма ввиду малого количества информации о сцене. Тем не менее многую информацию можно получить из статистики датасета: зачастую поверхности гладкие, цвета обычно однородные, а освещение естественное. Некоторые ограничения неощутимы на практике, например, в 2D-играх допустимо применять алгоритм, учитывающий лишь объекты области экрана, так как основные действия сосредотачиваются на экране.

1. Deep Shading: Convolutional Neural Networks for Screen-Space Shading / Oliver Nalbach [and others]. – Saarbrücken: Max Planck Institute for Informatics, 2016.
2. Alican Mertan. Single Image Depth Estimation: An Overview / Alican Mertan, Damien Jade Duff, Gozde Unal. – Istanbul Technical University. – 13.04.2021.
3. Tom van Dijk. How do neural networks see depth in single images? / Tom van Dijk, Guido de Croon // Technische Universiteit Delft, Delft, Netherlands. – 01.10.2019.
4. 3D Ken Burns Effect from a Single Image / Simon Niklaus [and others]. – ACM Transactions on Graphics, Vol. 38, No. 6, Article 184. – Nov.2019.

ПОДХОДЫ К ИДЕНТИФИКАЦИИ ТЕХНИЧЕСКОГО СОСТОЯНИЯ ВЕРХНЕГО СТРОЕНИЯ ЖЕЛЕЗНОДОРОЖНОГО ПУТИ

Владова А. Ю.

Лаборатория 41, Институт проблем управления им. В.А. Трапезникова РАН

Финансовый университет при Правительстве Российской Федерации

Москва, РФ

E-mail: {avladova}@mail.ru

Рассмотрены аналитический, статистический и обучающий (data science) подходы к оценке технического состояния верхнего строения железнодорожного пути как идентифицируемой системы. Представлены проблемы анализа технического состояния систем, достоинства и недостатки методов.

ВВЕДЕНИЕ

Верхнее строение железнодорожного пути — это система, в котором происходит взаимодействие между разнотипными признаками (инженерные, трибологические, климатические, геологические, гидрологические и др.) и формируются наблюдаемые сигналы. Система является динамической, в которой текущие значения наблюдаемых сигналов зависят не только от текущих, но и от более ранних значений внешних воздействий. Воздействия на систему в процессе эксплуатации доступны косвенной оценке по влиянию, оказываемую ими на выходные сигналы. Как следствие типового подхода к сбору информации выходные сигналы привязаны к дискретным моментам времени в общем случае распределенным неравномерно.

1. АНАЛИТИЧЕСКИЙ И СТАТИСТИЧЕСКИЙ ПОДХОДЫ

В теории идентификации связь между распределенным во времени внешним воздействием $u(t)$ и наблюдаемым сигналом $y(t)$ в линейной стационарной системе описывают импульсной реакцией (весовой функцией) $g(\tau)$:

$$y(t) = \int_0^{\infty} g(\tau)u(t - \tau)d\tau \quad (1)$$

Используя (1) можно провести оценку и прогнозирование наблюдаемого сигнала $y(t)$ если известно внешнее воздействие $u(t)$, что в большинстве случаев нереально. Всегда существуют помехи, влияющие на результат наблюдений: шум наблюдений — датчики подвергаются воздействию помех и дрейфу; неконтролируемые входы — в систему поступают сигналы, которые имеют характер входных, но не контролируются пользователем (например, сезонное промерзание рельс, подтопление). Линейная теория идентификации дает предположение, что влияние помех сводится к аддитивной компоненте $v(t)$ в наблюдаемом сигнале:

$$y(t) = \sum_{k=1}^{\infty} g(k)u(t - k) + v(t) \quad (2)$$

Наиболее характерной особенностью помехи является то, что ее значение заранее неизвестно. Однако информация о прошлых значениях помехи может использоваться для ее оценки в будущем. Для описания будущего поведения помехи принято использовать вероятностный подход. Однако реализация этого подхода требует [1]:

- формирования статистических гипотез о будущем поведении помех,
- задания различных функций плотности вероятности помех,
- анализ помех, являющихся смесью детерминированных и случайных компонент и пр.

Кроме того, для оценки технического состояния реальной системы в рамках теории идентификации часто необходим переход к системе нелинейных уравнений с запаздывающими аргументами [2], например, когда функция $g(t)$ заменяется на $g(t, t-k)$ и (2) записывается в виде:

$$y(t) = \sum_{s=-\infty}^{t-1} g(t, s)u(s) + v(t) \quad (3)$$

где $g(t, s)$ ($t=s, s+1, \dots$) - реакция на отклик в момент времени t на единичный входной импульс в момент s . С другой стороны, использование методов математической статистики в случае оценки технического состояния систем не всегда приводит к успеху. Одна из причин этого — концепция усреднения по выборке, приводящая к операциям над фиктивными величинами (например, средний пропущенный тоннаж, средний износ). Второй причиной является то, что погашение неточностей наблюдений при статистической обработке данных происходит в случае, если неточности являются результатом случайных погрешностей [3]. Если отклонения в данных носят неслучайный характер, то необходимо корректировать данные, либо собирать их заново. Третьей проблемой является то, что область применения статистических моделей ограничена ближайшей окрестностью наблюдаемых значений. Таким образом, проблемами анализа технического состояния систем являются [4]:

- проблема объема выборки или стохастической сходимости;
- проблема большого числа признаков (размерность модели);
- проблема многокомпонентного наблюдаемого сигнала. Один из способов решения – применение интегрирующей функции (например, функции полезности, эффективности или надежности);
- проблема нелинейности, когда одна из функций (2) или все три нелинейны.

II. DATA SCIENCE ПОДХОД

С другой стороны, современные системы мониторинга технического состояния объектов характеризуются тем, что емкости компьютерных хранилищ увеличились и стали дешевы, сенсоры уменьшились в размерах и способны отслеживать сигналы различной физической природы, а также передавать данные на значительные расстояния, а облачные вычисления на основе технологии анализа данных позволяют использовать возможности огромного количества машин для манипулирования этими данными. Технологии анализа данных при оценке технического состояния систем помогают решить следующие проблемы: выбор критерия селекции, списка признаков (размерности модели) и сложности модели – при синтезе модели методами машинного обучения сложный объект, имеющий сотни взаимосвязанных признаков, может быть сравнительно просто алгоритмически описан; выявление однотипных фрагментов многоаспектных взаимоотношений в данных. Эти фрагменты представляют собой закономерности, свойственные подвыборкам данных; тестирование полученных моделей: весь исследуемый массив данных разделяют на две неравные группы. Большая из них (обучающая) является исходным материалом для построения моделей, а меньшая представляет собой тестовую группу, на которой проверяются полученные модели. Критерием, по которому оценивается модель, является разность в точности оценки между группами. Один из подходов в рамках применения технологии Data Science предлагает статья [6], посвященная оценке воздействия деформированных колес на железнодорожный путь путем полноволнового численного моделирования на суперкомпьютерных системах. Для специалистов в сфере машинного обучения статья интересна тем, что перечисляет целый ряд факторов, влияющих на зависимую переменную, например, степень поврежденности колесной пары, наличие исходных дефектов в рельсе, методика укладки шпал, влияние пустот под шпалами и др. Авторы статьи [7] предлагают проводить классификацию режима работы устройств железнодорожной автоматизации и телемеханики (в частности, электри-

ческой рельсовой цепи, работающей в трех режимах) на основе логистической регрессии или метода опорных векторов с гауссовым ядром. При этом вектор признаков формируют по параметрам электрических сигналов на входе и выходе устройства. В статье [8] проведен многофакторный анализ и кластеризация данных по отступлениям и неисправностям пути, архивам капитальных ремонтов, рельсовым книгам, активах земляного полотна, ведомостям дефектных рельсах, выполнено статистическое исследование ряда признаков и установлена нелинейная взаимосвязь между ними.

III. ЗАКЛЮЧЕНИЕ

Анализ литературы показал, что сформированы подходы на основе статистических, вероятностных и обучающих методов по отдельным типам данных. Современные технологии анализа данных позволяют интегрировать математические и статистические методы с программированием и способами хранения данных. Для решения проблемы идентификации технического состояния верхнего строения пути предложено интегрировать данные различной физической природы (пропущенный тоннаж, паспортные характеристики рельс, особенности плана и профиля пути и др.).

IV. СПИСОК ЛИТЕРАТУРЫ

1. Льюнг, Л. Идентификация систем. Теория для пользователя /Л. Льюнг // М.: Наука. Гл. ред. физ.-мат. лит., 1991. - 432 с.
2. Принятие решений на основе самоорганизации /Ивахненко [и др.]. - Изд-во: М.: Советское радио, 1976. - 280 с.
3. Венецкий, И. Г. Основные математико-статистические понятия и формулы в экономическом анализе / И. Г. Венецкий, В. И. Венецкая // М.: Статистика, 1979. - 447 с.
4. Нейлор, Т. Машинные имитационные эксперименты с моделями экономических систем / Т. Нейлор - Изд-во: Мир, Москва, 1975. - 504 с.
5. Федосеев, А. А. Технология data mining в задачах прогнозирования развития транспортной инфраструктуры [Электронный ресурс] / Современные проблемы науки и образования. - 2013. № 1. - Режим доступа: <http://www.science-education.ru/ru/article/view?id=8153>. - Дата доступа: 06.06.2021.
6. Фаворская, А. В. Разработка математических моделей, численных методов и расчетных программ для выявления дефектов элементов системы «колесо-рельс» / Фаворская [и др.]. - Бюллетень Объединенного ученого совета ОАО РЖД. № 1, 2018. с. 49–63.
7. Prisukhina, I. V. Machine state classification of electric track circuit by means of support vector machine / I. V. Prisukhina, D. V. Borisenko // Omsk Scientific Bulletin, 2018. с. 126–130.
8. Владова, А. Ю. Применение технологии анализа данных для идентификации технического состояния верхнего строения пути /Труды 14-й Международной конференции «Управление развитием крупномасштабных систем». М.: ИПУ РАН, 2021 (в печати).

ОБЗОР МЕТОДОВ ГЕНЕРАЦИИ ТЕКСТА НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

Воронова В. В., Удовин И. А.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: veronika.voronova31@gmail.com, wilcot@ya.ru

В последние годы были изобретены и развиты многие подходы искусственного интеллекта для генерации различных данных: текста, изображений, музыки. Возникает желание понять, могут ли одни и те же методы быть переиспользованы для генерации другого типа данных. Для этого необходимо изучить конкретные методы, применяемые в каждом случае. В данной статье приводятся обзоры подходов, применяемых в области генерации текстов естественного языка, и алгоритмы, которые обеспечивают эти механизмы.

ВВЕДЕНИЕ

Одно из значительных достижений человека - способность передавать данные и делиться ими. При обсуждении языка, на котором говорят люди, возникает идея его генерации, ведь язык является одним из наиболее сложных умений человека. По некоторым оценкам, лишь 21 процент[1] существующей информации структурирован. Информация создается повсюду в больших масштабах в виде твитов и сообщений, где большая часть существует в текстовой форме, которая по своей природе очень неструктурирована. Для получения важных знаний из этой информации, важно познакомиться с системами обработки естественного языка(NLP). Обработка естественного языка - область разработки программного обеспечения и искусственного интеллекта, которая работает с человеческими языками. Это подходы, при которых мы автоматически описываем и анализируем язык.

I. ПРЕДСТАВЛЕНИЕ ДАННЫХ

Подходы распределенного представления входных данных являются фундаментальной идеей для глубоких генеративных моделей, особенно применительно к задачам обработки естественного языка. Нераспределенное представление делает данные разреженными, что неэффективно по нескольким причинам: размерность данных увеличивается по мере увеличения структуры, а так же повышается глубина генеративных моделей для достижения лучших результатов в задачах. Если набор данных содержит функции, которые имеют схожие значения, лучше получить представление, описывающее эти сходства. Векторное представление слов(word embeddings) - это отображение дискретной категориальной переменной в вектор непрерывных чисел.

Word2Vec - это метод, который предсказывает целевое слово из заданного контекста слов (модель Continuous Bag of Words (CBOW)) или предсказывает контекст из целевого слова (модель Skip-gram)[2]. Проблема Word2Vec в том,

что он полагается на локальный контекст предложений, что означает, что он захватывает только семантическую информацию языка. Еще один подход, Glove (Global Vectors), с другой стороны, улавливает как глобальный, так и локальный контекст словарного запаса при преобразовании слов в векторы. У каждого из них есть свои применения: Word2Vec очень хорошо справляется с задачами аналогии, а Glove работает с совпадением слов. Еще один метод представления слов в векторы - FastText - представляет каждое слово как n-граммы символов, а не содержит слова напрямую.

II. МЕТОДЫ, ОСНОВАННЫЕ НА РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЯХ

Рекуррентные нейронные сети(RRNs) - мощный алгоритм для задач обработки естественного языка, особенно при моделировании последовательных данных. Поскольку рекуррентные нейронные сети содержат внутреннюю память, благодаря которой они могут запоминать предыдущий ввод, а также текущий ввод, то это значительно упрощает задачи моделирования последовательности. Вывод на любом временном шаге зависит не только от текущего ввода, но и от вывода, сгенерированного на предыдущих временных шагах, что делает алгоритм очень подходящим для таких задач, как генерация языка, языковой перевод, анализ тональности и т. д. Но было замечено, что обучение таких сетей очень сложно, что препятствует их использованию во многих задачах.

Долгая краткосрочная память(LSTM) наследует ту же архитектуру, что и обычные рекуррентные нейронные сети без скрытого состояния. Единицы памяти в LSTM называются ячейками, которые принимают в качестве входных данных комбинацию предыдущего состояния и текущего ввода. Эти ячейки фактически решают, что оставить в памяти, а что удалить. GRU - это еще одно расширение стандартных рекуррентных нейронных сетей, которое изменяет архитектуру LSTM с помощью стробирующей сети, которая генерирует сигналы, которые управ-

ляют текущим вводом и предыдущей памятью, чтобы обновить текущую активацию и текущее состояние сети. Это проще, чем LSTM, в котором обновление параметров также используется для ячеек в соответствии с алгоритмом.

Было предложено множество глубоких генеративных моделей, в которых двунаправленные рекуррентные нейронные сети используются для генерации последовательности выходных данных. Идея таких сетей заключается в том, что выход на временном шаге может зависеть не только от предыдущих элементов, но и от будущих элементов последовательности. Они состоят из двух независимых рекуррентных нейронных сетей.

III. СВЕРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ

Сверточные нейронные сети - один из популярных алгоритмов, используемых для компьютерного зрения. Их исследования, применяемых для задач обработки естественного языка, были начаты недавно, и дали интересные результаты. Было предложено множество методов классификации текстов с использованием сверточных нейронных сетей. В отличие от задач компьютерного зрения, где в качестве входных данных используются пиксели изображения, в задачах языка вместо пикселей изображения используются предложения, слова или иногда символы, в зависимости от классификационной проблемы.

IV. ВАРИАЦИОННЫЕ АВТОКОДИРОВЩИКИ

Популярные модели глубокого обучения требуют большого количества структурированных данных. Маркировка неструктурированных данных занимает очень много времени. Один из способов решения этой проблемы - использовать обучение без учителя (unsupervised) для обучения на данных без меток. Вариационные автокодировщики (Variational Auto-Encoders) - одна из мощных генеративных моделей, работающих с немаркированными данными. Он содержит кодировщик, который кодирует данные в скрытые переменные, а так же декодер, который декодирует эти скрытые переменные для восстановления данных. Вариационные автокодировщики были разработаны как один из популярных способов обучения без учителя для сложных распределений. Их применение для генерации дискретных данных (текста) ограничено. Основная проблема использования для генерации текста - это коллапс KL (это означает, что когда декодер становится более мощным, чем цель обучения, может быть решена с помощью ложной стратегии). Были предложены некоторые методы, что-

бы преодолеть проблему, однако они не решили ее полностью.

V. ГЕНЕРАТИВНЫЕ СОСТЯЗАТЕЛЬНЫЕ СЕТИ

Мы научили машины разбираться в вещах, многие архитектуры глубокого обучения заслуживают признания за их творческий успех. Несмотря на это, многие глубокие генеративные модели не достигли большого успеха из-за их неспособности аппроксимировать трудноразрешимые вероятностные вычисления. Но было найдено решение, которое может обойти эти проблемы, называемое генеративные состязательные сети (Generative Adversarial Networks - GAN). GAN - это популярный алгоритм глубокого обучения, использующий состязательный подход, отличный от обычной нейронной сети. GAN содержит две модели, которые обучаются состязательным способом. Генератор производит выборки данных, дискриминатор - классифицирует эти выборки данных как реальные (обучающие данные) или фальшивые (сгенерированные генератором). Цель генератора - производить выборки, которые очень близки к истинным данным, чтобы можно было обмануть дискриминатор, а цель дискриминатора - точно классифицировать эти два типа выборок данных. Дискриминатор пытается максимизировать целевую функцию, а генератор пытается минимизировать целевую функцию.

VI. ЗАКЛЮЧЕНИЕ

Для генерации текста естественного языка могут использоваться как методы обучения с учителем, так и без него. Все методы имеют как достоинства, так и слабые места, и чаще всего хороши в решении конкретных задач. Так же необходимо иметь в виду, что некоторые решения могут быть ограничены вычислительными ресурсами.

СПИСОК ЛИТЕРАТУРЫ

1. The survey: Text generation models in deep learning [Электронный ресурс] / sciencedirect.com - Режим доступа: <https://www.sciencedirect.com/science/article/pii/S1319157820303360/>. - Дата доступа: 18.10.2021.
2. Word2Vec Explained [Электронный ресурс] / towardsdatascience.com - Режим доступа: <https://towardsdatascience.com/word2vec-explained-49c52b4ccb71/>. - Дата доступа: 18.10.2021.
3. A Gentle Introduction to Generative Adversarial Networks [Электронный ресурс] / machinelearningmastery.com - Режим доступа: <https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/>. - Дата доступа: 18.10.2021.

РАЗРАБОТКА ИМИТАЦИОННОЙ МОДЕЛИ ВИРТУАЛЬНОЙ ЧАСТНОЙ СЕТИ ЭЛЕКТРОСВЯЗИ СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ В СЕТЕВОМ СИМУЛЯТОРЕ NS-3

Врублевский С. С., Бысов А.А.
Кафедра связи, Военная академия Республики Беларусь
Минск, Республика Беларусь
E-mail: sergeyvrublevsky0@gmail.com

В докладе представлена имитационная модель виртуальной частной сети электросвязи специального назначения, которая позволяет проводить исследования виртуальных частных сетей.

ВВЕДЕНИЕ

Виртуальная частная сеть (Virtual Private Network VPN) представляет собой выделенную сеть передачи данных, построенную на инфраструктуре телекоммуникационной сети общего пользования, в которой конфиденциальность и защищенность информации пользователя обеспечивается механизмами туннелирования и средствами информационной безопасности [1].



Рис. 1 – Фрагмент СЭСН с VPN-туннелем.

I. ОСНОВНАЯ ЧАСТЬ

Виртуальные частные сети подразделяются на:

- сети, построенные на оборудовании, которое устанавливается на стороне клиента и служит для его подключения к сети провайдера;
- сети, построенные на оборудовании, которое устанавливается на стороне провайдера.

И те, и другие подразделяются на три класса в зависимости от принципа организации связи пользователей сети:

- ведомственные (внутрикорпоративные) сети VPN – как правило, строятся на собственной сетевой инфраструктуре без использования ресурсов сети связи общего пользования;
- межведомственные (межкорпоративные) сети VPN – используют как собственную сетевую инфраструктуру, так и инфраструктуру сети провайдера;
- сети VPN удаленного доступа – данный класс сетей VPN предполагает подключение пользователя к сети VPN при помощи специального аппаратного (криптомаршрутизаторы) и программного (Cisco

AnyConnect Secure Mobility Client, Avast SecureLine VPN) обеспечения.

Виртуальные частные сети могут быть реализованы на базе протоколов модели OSI на следующих уровнях:

- канальный – L2VPN (L2TP, PPTP, VPLS, VPWS);
- сетевой – L3VPN (IPsec, GRE, BGP/MPLS, VPRN);
- сеансовый – L5VPN [2].

Виртуальные частные сети, реализованные на базе протоколов третьего уровня, получили наибольшее распространение. Это обусловлено центральной ролью протокола IP в стеках протоколов модели OSI.

Сеть электросвязи специального назначения (СЭСН) – это сеть, предназначенная для обеспечения нужд государственного управления, национальной безопасности, обороны, охраны правопорядка, предупреждения и ликвидации чрезвычайных ситуаций [3]. Применение технологии VPN получило широкое распространение в СЭСН различных государств по следующим причинам:

- применение стандартных стеков протоколов (например, TCP/IP);
- использование средств IP-шифрования, для построения защищенных каналов (туннелей VPN);
- необходимостью разграничения трафика в зависимости от его класса и приоритета пользователей.

Для создания надежно функционирующей СЭСН необходимо еще на этапе проектирования знать возможные характеристики узлов сети, ввиду того что современный мультисервисный трафик, циркулирующий в сети обладает свойствами самоподобия, который не поддается строгому математическому анализу. Основным инструментом анализа может являться имитационное моделирование без использования реального сетевого оборудования.

Одним из средств имитационного моделирования компьютерных сетей является сетевой симулятор Network Simulator 3 (NS-3)[4]. Данная среда моделирования представляет собой си-

мулятор сети связи с дискретными событиями, предназначенный для исследований и использования в образовательных целях. Поддерживает большой стек протоколов и позволяет моделировать компьютерные сети с различными топологиями. Является бесплатным программным обеспечением с открытым исходным кодом (C++ / Python), а также работает с внешними инструментами анимации, анализа данных (создает файлы формата .pcap для работы с Wireshark, а также трейс-файлы в формате ASCII) и визуализации (NetAnim).

Для имитации VPN-туннеля (далее – туннель) вида «IP-over-IP» была создана модель фрагмента СЭСН (рис. 2) с помощью симулятора NS-3 [5]. Этапы моделирования в NS-3 представлены на рис. 3.

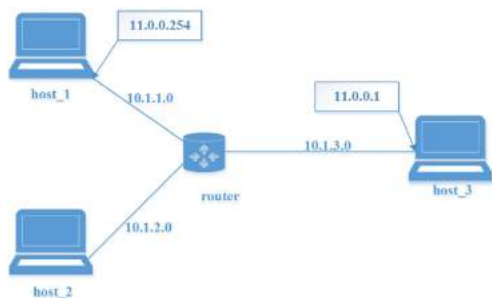


Рис. 2 – Фрагмент СЭСН с VPN-туннелем.

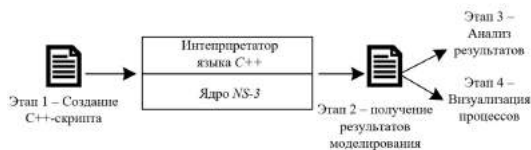


Рис. 3 – Этапы моделирования фрагмента СЭСН с помощью NS-3

Каждое из сетевых устройств имеет свой IP-адрес. Но при создании туннеля им присваиваются виртуальные адреса. Для этого был реализован класс Tunnel, в котором реализуется данная функция. Для проверки функционирования имитационной модели был передан поток UDP трафика от host3 передается с виртуального адреса 11.0.0.1 на host1 с виртуальным адресом 11.0.0.254, что видно из рис. 4.

```

# 1.00914 /ModelList/0/DeviceList/1/NS3::PointToPointNetDevice/TxQueue/Enqueue
ns3::PppHeader (Point-to-Point Protocol: IP (0x0021)) ns3::Ipv4Header (tos 0x0
DSCP Default ECN Not-ECT ttl 64 id 0 protocol 17 offset (bytes) 0 flags [none]
length: 568 10.1.1.1 > 10.1.3.1) ns3::UdpHeader (length: 548 667 > 667)
ns3::Ipv4Header (tos 0x0 DSCP Default ECN Not-ECT ttl 64 id 0 protocol 17
offset (bytes) 0 flags [none] length: 540 11.0.0.1 > 11.0.0.254)
ns3::UdpHeader (length: 520 49153 > 9) Payload (size=512)

```

Рис. 4 – Событие моделирования, показывающее передачу потока трафика между виртуальными адресами.

Для визуализации процесса передачи потока трафика была использована среда NetAnim, что видно из рис. 5

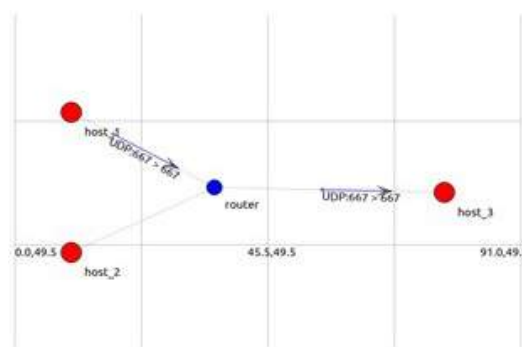


Рис. 5 – Визуализация передачи потока трафика в среде NetAnim.

II. ЗАКЛЮЧЕНИЕ

Построенная имитационная модель позволяет проводить исследование сетей VPN и является гибким инструментом для анализа способов их планирования. Разработанный класс Tunnel для фрагмента сети позволит создать модель для полной СЭСН с VPN. Это и является дальнейшим направлением исследования

III. СПИСОК ЛИТЕРАТУРЫ

1. Mitra, D. Virtual Private Networks: Joint Resource Allocation and Routing Design / D. Mitra, J. A. Morrison, K. G. Ramakrishnan // Proc. of the 18th Annual Joint Conference of the IEEE Computer and Communications Societies. – 1999. – P. 480–490.
2. Сапрыкин, А. В. Исследование и разработка методов анализа вероятностно-временных характеристик узлов сетей связи специального назначения: автореф. дисс. ... канд. тех. наук: 05.12.13 / А. В. Сапрыкин Поволжский гос. ун-т телекоммуникаций и информатики. – Самара, 2008. – 16с.
3. Об электросвязи : Закон Респ. Беларусь от 19.05.2005г. №45-3 : в ред. от 06.08.2007г. №277-3 : с изм. и доп. от 24.05.2021г. №109-3. – Минск, 2021. – 34с.
4. NS-3 Model Library [Electronic resource] / – Mode of access: <http://www.nsnam.org/docs/models/ns-3-model-library>. – Date of access: 05.10.2021.
5. NS-3 Manual [Electronic resource] / – Mode of access: <http://www.nsnam.org/docs/manual/ns-3-manual.pdf> – Date of access: 05.10.2021.

МЕТОДЫ АНАЛИЗА ФИНАНСОВОГО РЫНКА С ПРИМЕНЕНИЕМ НЕЙРОСЕТЕВОГО ПОДХОДА

Дворниченко А. А.

Кафедра интеллектуальных информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: leshadvornichenko@mail.ru

ВВЕДЕНИЕ

В потреблении, выходящем за рамки борьбы за физическое существование, в различной степени участвует подавляющее большинство населения. Повышение средней заработной платы человека или платы за предоставленные услуги влечет за собой рост расходов на потребляемые товары или услуги и, следовательно, рост количества финансовых операций. Для сохранения или улучшения финансового состояния человека и в условиях неопределенности финансового рынка существует необходимость анализировать набор факторов, оказывающих влияние на финансовый рынок. Ввиду роста количества таких факторов и скорости их изменения, человек становится не в состоянии быстро и точно принимать решения в данной области. Этим обуславливается рост спроса на технологии, позволяющие контролировать или, как минимум, предугадывать состояние рынка в обозримом будущем. Существует множество решений для таких проблем, порождаемых спросом, к которым, в частности, относится внедрение соответствующего программного обеспечения. Однако, данные решения имеют строго описанный алгоритм работы, что делает их неэффективными в условиях постоянного изменения рынка. Использование нейросетевого подхода в качестве метода анализа экономических данных позволяет производить расчеты и принимать решения, не опираясь на фиксированное количество входных данных. В данной статье будет рассмотрен данный подход.

I. ПРЕДЛАГАЕМЫЙ ПОДХОД К АНАЛИЗУ ПОКАЗАТЕЛЕЙ ФИНАНСОВОГО РЫНКА

Среди нейросетевых технологий, применяемых в анализе финансового рынка, широкое распространение получили искусственные нейронные сети (ИНС). ИНС – это программно- или аппаратно-реализованные системы, построенные на взаимодействии формальных нейронов – математических моделей биологических нейронов.

Базовая структура ИНС может быть проиллюстрирована как $Y = F * (X^T * w + c)$ относительно независимых (входных) переменных X , весовых членов w и постоянных членов c . Y – зависимая переменная, а X формируется как матрица размера $n * t$ для количества обуча-

ющей выборки n и количества входных переменных t . Чтобы применить эту структуру в финансах, Y можно рассматривать как уровень кредитного риска клиентов, курса акций или доходность портфеля. F – это функция активации, которая уникальна и отличается от регрессионных моделей. F обычно формулируется как сигмовидные функции и tanh-функции. В результате объединения нескольких перцептронов в каждом слое и добавления скрытого слоя от Z_1 до Z_4 в середине, получается многослойная ИНС, где входными слоями будут X_s , а выходными слоями – Y_s . Более того, несколько Y_s также применимы: например, управляющих фондами часто волнуют будущие цены и их колебания. Достоинством применения таких сетей в процессе принятия решения является неограниченное количество нейронов в каждом из слоев, что позволяет учесть неограниченное количество факторов. Именно наличие скрытых слоев позволяет выявлять сложные, нелинейные взаимосвязи между входными и выходными данными. На рисунке 1 проиллюстрирована базовая структура ИНС [1].

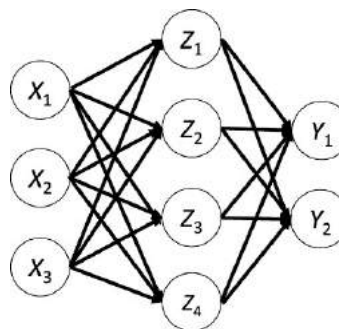


Рис. 1 – Базовая структура искусственной нейронной сети

Основываясь на базовой структуре ИНС, показанной на рисунке 1, традиционные нейронные сети делятся на следующие виды:

- Глубокая нейронная сеть (ГНС);
- Рекуррентная нейронная сеть (РНС);
- Нейронная сеть прямого распространения (НСПР).

В свою очередь, ГНС – это ИНС с несколькими скрытыми слоями между входным и выходным слоями. ГНС находит корректный метод математических преобразований, чтобы превратить входные данные в выходящие, независимо от линейной или нелинейной корреляции ГНС,

как правило, представляют собой сети с прямой связью, в которых данные передаются от входного уровня к выходному уровню без обратной связи. Сначала ГНС создает карту виртуальных нейронов и назначает случайные числовые значения или «веса» соединениям между ними. Веса и входные данные умножаются и возвращают выходной сигнал от 0 до 1. Если сеть не точно распознала конкретный шаблон, алгоритм будет корректировать весовые коэффициенты. Таким образом, алгоритм может сделать определённые параметры более значимыми, пока он не определит правильные математические манипуляции для полной обработки данных [2].

РНС – ИНС, в которой связи между элементами образуют направленную последовательность, благодаря чему появляется возможность обрабатывать серии событий во времени или последовательные пространственные цепочки. В отличие от ГНС, РНС могут использовать свою внутреннюю память для обработки последовательностей произвольной длины [3].

НСПР – ИНС, в которых сигнал распространяется строго от входного слоя к выходному. В обратном направлении сигнал не распространяется. Использование этих моделей может игнорировать порядок данных и значение времени [4].

Как представлено на рисунке 2, РНС имеет новую структуру ИНС, которая может решать проблемы долгосрочной зависимости и порядка между входными переменными. Поскольку финансовые данные во временных рядах очень распространены, выявление скрытых корреляций имеет решающее значение в реальном мире. РНС может лучше решать эту проблему по сравнению с другими методами «скользящего среднего», свойственных для нейронных сетей других типов.

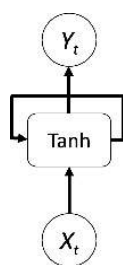


Рис. 2 – Структура рекуррентной нейронной сети

Хотя РНС может решить проблему порядка временных рядов, проблема долгосрочных зависимостей остается. Подобрать оптимальный вес для долгосрочных данных сложно. ИНС типа «Долгая краткосрочная память», как подтип РНС, имеют закрытую ячейку для преодоления долгосрочных зависимостей путем комбинирования различных функций активации (например, сигмовидной и tanh). На рисунке 3 представлена нейронная сеть типа «Долгая краткосрочная память» [5].

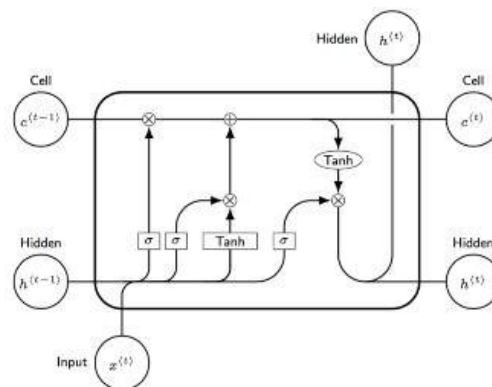


Рис. 3 – Структура нейронной сети типа «Долгая краткосрочная память»

Одним из наиболее эффективных методов обучения ИНС, представленных выше, является глубокое обучение (ГО) — совокупность методов машинного обучения (с учителем, с частичным привлечением учителя, без учителя, с подкреплением), основанных на обучении представлениям, а не специализированных алгоритмах под конкретные задачи. По сравнению с традиционными методами машинного обучения, такими как машина опорных векторов и k -ближайших соседей, ГО обладает преимуществами неконтролируемого обучения функций, сильной возможностью обобщения и надежной обучающей способностью на основе больших данных [6].

Таким образом, применение ИНС, обученных с помощью алгоритмов ГО, в различных областях финансового сектора позволяет получить различные преимущества, в зависимости от типа ИНС. Например, для прогноза состояния фондового рынка наибольшее распространение получила ИНС типа «Долгая краткосрочная память», которая позволяет извлекать информацию из новостей рынка и строить глубокую нейронную генеративную модель для прогнозирования движения цены акций. Для оценки риска банковского дефолта используются ГНС, позволяющие выявить нелинейные взаимосвязи между факторами.

II. СПИСОК ЛИТЕРАТУРЫ

1. Rashid, T. R. Make Your Own Neural Network 1 / T. R. Rashid// – 2016. – Vol. 342, № 10. – P. 113–125.
2. Хайкин, С. Х. Нейронные сети: полный курс / С. Х. Хайкин// – 2006. –СПб.: Синергия, 2006. – 84–89 с.
3. Хайкин, С. Х. Нейронные сети: полный курс / С. Х. Хайкин// – 2006. –СПб.: Синергия, 2006. – 112–121 с.
4. Хайкин, С. Х. Нейронные сети: полный курс / С. Х. Хайкин// – 2006. –СПб.: Синергия, 2006. – 64–75 с.
5. Deep learning in finance and banking: A literature review and classification [Электронный ресурс] / SpringerOpen. – Режим доступа: <https://fbr.springeropen.com/articles/10.1186/s11782-020-00082>. – Дата доступа: 10.10.2021.
6. Бенджио, И. Б. Глубокое обучение / И. Б. Бенджио// – 2016. – СПб.: Инкарт, 2016, 2017. – 28–33 с.

ГЕНЕТИЧЕСКИЙ АЛГОРИТМ ДЛЯ РАСКРОЯ ЛИСТОВЫХ МАТЕРИАЛОВ

Денисевич Д. А., Якимов Е. А.

Кафедра «Автоматизированные системы управления», Белорусско-Российский университет

Могилев, Республика Беларусь

E-mail: denis1351@yandex.ru, e-soft@bk.ru

Рассматривается генетический алгоритм для раскроя листового материала прямоугольных форм. Предложен способ сравнения алгоритмов при решении задач одномерного раскроя по критерию «бесполезный материал». Представлены результаты исследования трех алгоритмов: первый подходящий, первый подходящий с упорядочиванием и генетический алгоритм.

ВВЕДЕНИЕ

Задача раскроя листовых материалов является проблемно-ориентированной в компьютерных информационных системах и сводится к задаче поиска оптимальных решений. Технология раскроя часто регламентируется специальной технической документацией в виде раскройных карт, которые поступают на раскройный участок в специальном формате. Особенности проектирования карт раскроя является нелинейность, многоэкстремальность, высокая размерность пространства поиска. Выполняется с учетом наиболее эффективного использования площади листового материала и минимизации отходов производства.

I. ГЕНЕТИЧЕСКИЙ АЛГОРИТМ

Генетический алгоритм – это алгоритм, который используется для решения задач оптимизации и моделирования путём случайного подбора популяции, индивидумов в популяции, хромосом, где определение оптимального решения основано на комбинировании (скрещивании генов) и вариации параметров (мутации) с использованием механизмов, аналогичных естественному отбору в природе (см. рис. 1).



Рис. 1 – Пример работы генетического алгоритма

В алгоритме применяются следующие понятия для формализации задачи. Гены – изделия,

получаемые из листового материала. Хромосомы – набор генов с известным расположением на листовом материале. Популяция – набор хромосом с генами, отличающие между собой расположением на заготовке.

В генетическом алгоритме начальная популяция формируется случайным образом. Каждому гену в хромосоме устанавливаются случайные координаты в прямоугольнике, площадь которого увеличена на половину суммы генов по длине и ширине от заданной пользователем. Такой подход объясняется генерацией большой численности популяции.

Задача нахождения оптимального решения, отбор хромосом и сравнение популяций реализуется с помощью функции приспособленности [1]. Она принимает неотрицательные значения для определения критерия, благодаря которому оценивается эффективность метода решения задачи раскроя листовых материалов.

В классической постановке задачи основным критерием оценки эффективности алгоритмов получения карт раскроя является максимальное значение коэффициента использования материала – отношение суммы площадей полученных изделий к площади исходного листового материала. Однако при использовании одного листового материала и одинакового набора изделий коэффициент будет равен во всех случаях. Поэтому предложен способ сравнения алгоритмов при решении задач одномерного раскроя по критерию «бесполезный материал».

Пусть $N = \{N_1, N_2, \dots, N_n | n = |N|\}$ – множество изделий, которое требуется изготовить из листового материала площадью S . Каждое изделие N_i , $i = 1, \dots, |N|$ имеет ширину W_i , длину L_i , площадь $s_i = W_i * L_i$.

Критерий «бесполезный материал» P – разница площади прямоугольника, охватывающего все изделия на листовом материале, и суммы площадей изделий на листовом материале [2] (рис. 2).

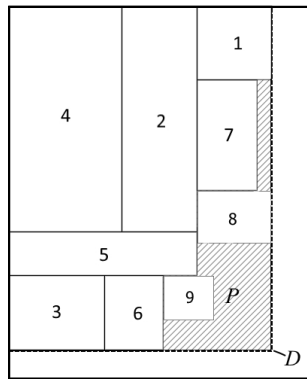


Рис. 2 – Карта раскроя листового материала: P – «бесполезный материал»; D – прямоугольник, охватывающий все изделия

Критерий «бесполезный материал» P вычисляется по формуле:

$$P = (D - \sum_{i=1}^{|N|} s_i * x_i) \rightarrow \min, i = 1, \dots, |N| \quad (1)$$

где x_i – количество изделий i -го типа, размещаемых на листовом материале

На этапе отбора хромосомы сортируются по убыванию согласно (1), половина из которых удаляется из первоначальной популяции.

Обмен генами между хромосомами осуществляется на этапе скрещивания, согласно случайной стратегии. Стратегии включают в себя четные гены, нечетные, первую половину генов относительно начала листового материала и вторую. Из пары хромосом выбирается та, в которой критерий P меньше.

Изменение положения генов в хромосоме реализуется на этапе мутации и только в половине случайных хромосом в популяции. Случайным образом выбирается ген для которого определяются возможные координаты для перемещения на листовом материале относительно начала координат. Возможными вариантами перемещения выбираются координаты такие, которые не включены в область других генов в хромосоме. Среди таких случайным образом выбираются координаты для размещения гена.

Для нахождения подходящего решения к аналитически неразрешимым или сложнорешаемым проблемам через отбор, скрещивание искомым характеристикам с использованием механизмов, напоминающих биологическую эволюцию, предложено сравнить популяции по критерию (1). Если сумма критерия хромосом первоначальной популяции больше суммы получившейся, то производим замену первой популяции на вторую и продолжаем эволюцию до тех пор, пока значение суммы критерия новой популяции будет больше предыдущей. Отбор наилучшего решения определяем снова с помощью функции приспособленности и полагаем, что последняя хромосома в популяции является искомым картой раскроя листового материала.

II. МЕТОДИКА И РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ

Для исследования эффективности генетического алгоритма предложено провести три эксперимента со структурными алгоритмами «Первый подходящий», «Первый подходящий с упорядочиванием», изделия в которых добавляются поочередно на карту раскроя начиная с угла листового материала. Если изделие не помещается в одну линию рядом с предыдущим, тогда отображается над ним при условиях ограничения по ширине и длине. Идея алгоритма «Первый подходящий с упорядочиванием» аналогичная, за исключением сортировки всех изделий по убыванию площади перед добавлением их на листовую материал.

В эксперименте 1 заказаны изделия с одинаковыми размерами. Изделия укладываются на листовом материале без отходов. В эксперименте 2 заказаны 7 изделий с размерами, отличающимися не более чем в 2 раза. В эксперименте 3 заказаны 15 изделий со значительными отличиями по размерам до 5 раз.

В первом эксперименте алгоритмы уложили все изделия на листовом материале без отходов. При этом генетический алгоритм значительно проигрывает по времени, $T = 12,77$.

Результаты исследования двух других экспериментов приведены в таблице 1.

Таблица 1 – Результаты исследования алгоритмов

Алгоритмы	Эксперимент 2		Эксперимент 3	
	T, c	P, m^2	T, c	P, m^2
Первый подходящий	0,006	1,75	0,006	2,29
Первый подходящий с упорядоч.	0,006	1,75	0,006	1,74
Генетический	7,99	1,12	78,81	1,07

ЗАКЛЮЧЕНИЕ

Выполненные исследования показывают, что генетический алгоритм по быстродействию уступает исследуемым алгоритмам и превосходит их по критерию «бесполезный материал». Однако в производственных условиях критерий быстродействия в поиске лучшего решения не является приоритетным для раскроя листовых материалов.

1. Хлопцев, А. А. Генетический алгоритм для оптимизации подбора параметров торговых ботов / А. А. Хлопцев, Ю. В. Кузьма // Информационные технологии и системы 2020 : материалы междунар. науч. конф., (Республика Беларусь, Минск, 18 ноября 2020 года) редкол.: Л. Ю. Шилин [и др.]. – Минск : БГУИР, 2020 – С. 88–89.
2. Демиденко, О. М. Исследование алгоритмов рационального раскроя листового материала / О. М. Демиденко, Е. А. Якимов, Д. А. Денисевич // Проблемы физики, математики и техники. – 2020. – № 1 (42). – С. 91–94.

ОНТОЛОГИЧЕСКИЙ ПОДХОД К ИНТЕГРАЦИИ СТОРОННИХ ФУНКЦИОНАЛЬНЫХ ПРИЛОЖЕНИЙ В СИСТЕМЫ ОБРАБОТКИ ЗНАНИЙ

Загорский А. Г.

Кафедра интеллектуальных информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: alexandr_zagorskiy@mail.ru

В данной работе описан подход к интеграции сторонних функциональных приложений, которые предоставляют собственный API, в системы, основанные на графодинамической обработке знаний. Также описан процесс интеграции сторонних функциональных веб-приложений.

ВВЕДЕНИЕ

Количество интеллектуальных систем (ИС), основанных на обработке знаний, незамедлительно растёт с каждым годом. Однако, человечество успело накопить уже значительный опыт в виде различных программных продуктов, основанных на стандартном подходе обработки информации. Задача интеграции таких продуктов в интеллектуальные системы с целью последующего их использования является ключевой и привлекает всё больше внимания. Возможность интегрировать в систему уже функционирующую подсистему позволяет экономить как значительное количество времени, так и ресурсы.

В данной работе будет рассмотрен подход к интеграции сторонних функциональных приложений в системы, построенные на основе семантических технологий.

I. ПОДХОД К ИНТЕГРАЦИИ СТОРОННИХ ФУНКЦИОНАЛЬНЫХ ПРИЛОЖЕНИЙ

Под функциональным приложением понимается программно реализованный механизм преобразования данных в соответствии с заданной функцией. Функциональность приложения может быть абсолютно любой: прогноз погоды в заданном городе, смена размеров изображений, синтез речи, и т.д. Зачастую такое приложение может предоставить программный интерфейс (API – Application Programming Interface), в котором описаны всевозможные форматы входов и соответствующих им выходов.

Основной проблемой интеграции сторонних функциональных приложений является различие форматов данных, с которыми работают приложения. Даже если два разных приложения предполагают обработку данных из одной предметной области (ПрО), с высокой вероятностью формат данных у них будет различаться.

Для интеграции стороннего функционального приложения в систему обработки знаний выделяются следующие глобальные задачи:

- подготовить базу знаний (БЗ) ИС;

- реализовать агент обработки знаний, который использует стороннее функциональное приложение, для решателя задач (РЗ) ИС. Подготовка БЗ подразумевает формализацию и наполнение знаниями ПрО, которая содержит необходимые знания для формирования запросов к стороннему функциональному приложению. Также необходимо формализовать ПрО, в которую будут погружаться полученные после обработки данные в виде знаний.

Обобщённый алгоритм агента обработки знаний с использованием сторонних приложений выглядит следующим образом:

1. извлечение из БЗ необходимых структур знаний;
2. преобразование извлечённых знаний в формат, необходимый для подачи на вход стороннего приложения;
3. отправление запроса и ожидание ответа стороннего приложения;
4. формирование конструкции знаний по полученным данным;
5. погружение новых знаний в БЗ ИС.

Проектирование агента предлагается производить на основе шаблона программного проектирования «декоратор» [1].

Таким образом, задача интеграции сводится к формализации структур входа и выхода конкретных функциональных приложений, а также реализации механизма отправки исходных данных и получения уже обработанных данных.

II. ОБРАБОТКА ЗНАНИЙ С ИСПОЛЬЗОВАНИЕМ СТОРОННИХ ВЕБ-ПРИЛОЖЕНИЙ

Распространённым видом функциональных приложений являются веб-приложения. Зачастую общение с такими приложениями реализуется на основе HTTP-протокола [2].

Для интеграции веб-приложений предлагается использовать многоагентную структуру. Диаграмма компонентов агентов обработки знаний с использованием сторонних веб-приложений представлена на рисунке 1.

Рассмотрим подробнее основные компоненты структуры:

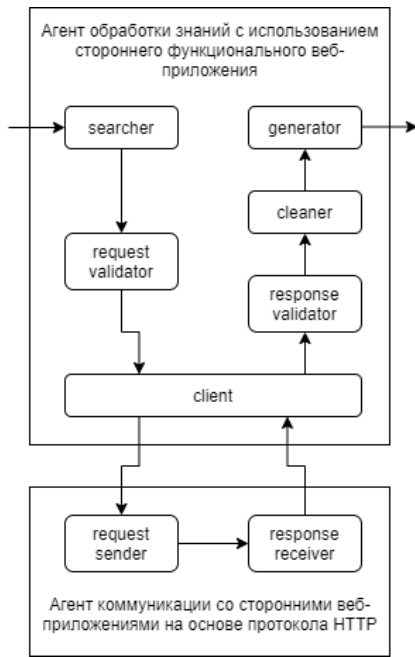


Рис. 1 – Основные компоненты агентов обработки знаний с использованием сторонних веб-приложений

- компонент *searcher* производит поиск необходимых конструкций знаний в БЗ по заданным шаблонам;
- компонент *request validator* проверяет полученные значения;
- компонент *client* формирует запрос для сервера, передаёт его агенту коммуникации со сторонними веб-приложениями на основе протокола HTTP, после чего ожидает ответа агента;
- компонент *response validator* обрабатывает полученные данные ответа сервера;
- компонент *cleaner* удаляет из памяти БЗ временные конструкции;
- компонент *generator* формирует структуры знаний по заданным шаблонам с последующей интеграцией в БЗ.

Агент коммуникации со сторонними веб-приложениями на основе протокола HTTP реализует механизм общения с серверами веб-приложений. Фрагмент БЗ в виде SCg-кода, содержащий примеры входных и выходных конструкций агента, представлены на рисунках 2 и 3 соответственно.

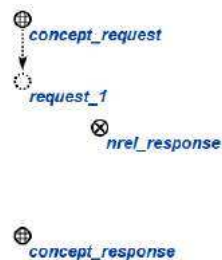


Рис. 2 – Пример входной конструкции агента коммуникации со сторонними веб-приложениями на основе протокола HTTP

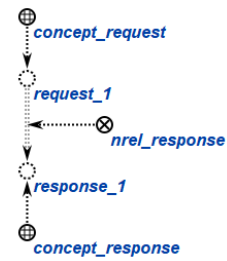


Рис. 3 – Пример выходной конструкции агента коммуникации со сторонними веб-приложениями на основе протокола HTTP

Функционирование агента базируется на множествах *concept_request* и *concept_response*. Вызов агента происходит при создании выходной дуги из множества запросов. Он извлекает полученный на входе запрос, отправляет его и ожидает ответа сервера. При получении ответа, он заносится во множество ответом БЗ, а также соединяем его с соответствующим запросом отношением. Так как информация после отработки агента извлекается моментально, после чего система формирует новые знания, не имеет смысла хранить и накапливать эти конструкции.

Такая структура агентов позволяет следовать принципу единой ответственности [3]: реализация первого агента является интеграцией конкретных функций посредством формализованных входов и выходов веб-приложения, в то время как агент отправки запросов является унифицированным средством коммуникации с любыми серверами сторонних веб-приложений. Также структура является легко расширяемой, что позволит, например, в будущем добавить механизм ограничения времени на запросы.

Аналогичным способом могут быть спроектированы агенты обработки знаний с использованием сторонних локальных функциональных приложений.

ЗАКЛЮЧЕНИЕ

Сформулированный подход к интеграции сторонних функциональных приложений, а также разработанная методология проектирования агентов обработки знаний с использованием сторонних функциональных веб-приложений позволяют ускорить процессы проектирования и разработки решателей задач интеллектуальных систем, основанных на технологии OSTIS [4].

1. Decorator [Electronic resource] – Mode of access: <https://refactoring.guru/design-patterns/decorator>. – Date of access: 18.10.2021
2. RFC 7230: HTTP/1.1 Message Syntax and Routing [Electronic resource] – Mode of access: <https://datatracker.ietf.org/doc/html/rfc7230>. – Date of access: 18.10.2021
3. Robert, M. SRP: The Single-Responsibility Principle. / Robert, M. // Agile Software Development, Principles, Patterns, and Practices – 2003. – pp. 95-98.
4. Семантические технологии проектирования интеллектуальных систем и семантические ассоциативные компьютеры / В. В. Голенков [и др.] // Доклады БГУИР. - 2019. - № 3 (121). - С. 42 - 50.

МОДЕЛИРОВАНИЕ РЕШЕНИЯ ЗАДАЧ ПРИКЛАДНЫХ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ НА ИНТЕГРАЦИОННОЙ ПЛАТФОРМЕ

Ивашенко В. П.

Кафедра интеллектуальных информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: ivashenko@bsuir.by

В работе рассматривается подход к моделированию решения задач прикладных интеллектуальных систем на основе интеграционной платформы, использующей унифицированное семантическое представление знаний и имеющей архитектуру системы, управляемой знаниями. Уточнены основные принципы и типология компонентов для реализации агентов решения задач.

ВВЕДЕНИЕ

Способность к решению задач [1] является одним из ключевых признаков интеллекта и интеллектуальных систем. Развитость интеллекта определяется множеством задач, которые способна решать система. Модель решения задач [1] определяет операции, которые можно применять для решения задач, множество возможных решений и множество задач, для которых рассматриваются эти решения. Рассмотрим подход к решению задач прикладных интеллектуальных систем на интеграционной платформе (ИП) [2], направленный на приобретение механизмов решения задач путём обучения и их интеграцию.

I. ОСНОВНЫЕ ПРИНЦИПЫ И КОМПОНЕНТЫ РЕШЕНИЯ ЗАДАЧ

Решение задач на ИП предполагает соотношение решаемых задач с общей классификацией задач [1] и использование интегрированных моделей решения задач. Кроме этого, при решении задач требуется соблюдение трёх принципов [1–2] обработки знаний на ИП: (1) учёт НЕ-факторов; (2) семантическое протоколирование; (3) обработка потока знаний (ОПЗ).

Учёт НЕ-факторов предполагает предпочтение нечёткого чёткому, неактуальному актуальному, временного постоянному при представлении, но при обработке – различаемого неразличаемому, и ориентирован не на работу с элементами, но – на работу со вхождением элементов в полностью представленные множества. Вследствие предпочтения различаемых вхождений неразличаемым более упорядоченные структуры предпочтительнее менее упорядоченных, ориентированные множества предпочтительнее неориентированных.

Семантическое протоколирование предполагает обработку не экземпляров данных, но – процессов становления данных-событий, поддерживая представление информации о повторяющихся и эмерджентных событиях.

ОПЗ предполагает выстраивание схем массовой параллельно-распределённой (асинхронно-

синхронной) обработки знаний по мере их готовности (управление от потока знаний) с целью решения независимых (обобщённых) задач.

Основными базовыми типами обрабатываемых знаний являются: множества, отношения и строки, представленные в графовой памяти в виде sc-цепочек или аналогичных структур [2].

Решение задач сводится к решению задач программными компонентами интеллектуальных систем, которые также классифицируются в соответствии с классификацией решаемых ими задач в рамках общей классификации задач.

В рамках архитектуры ИП основное решение задач связано с работой репрезентативных компонентов (РК) (не связанных с выводом через интерфейс машиной обработки образов и других клиентов), тогда как коммуникативные компоненты решают задачи, связанные с выводом через интерфейс машины обработки образов.

Среди РК можно выделить:

- когнитивностные РК (КРК);
- перформативностные РК (ПРК).

Компоненты ОПЗ, являясь сами по себе когнитивными компонентами, на входе получают одно состояние, а на выходе – возвращают другое состояние.

Среди КРК ОПЗ можно выделить:

- проверки (сверки) потока знаний;
- выбора потока знаний;
- поиска потока знаний.

Среди ПРК ОПЗ можно выделить:

- деструкции потока знаний;
- реконструкции потока знаний;
- конструкции потока знаний.

К необратимым невозвратным относят компоненты проверки и деструкции. К возвратным относятся компоненты выбора и реконструкции. К обратимым невозвратным относятся компоненты поиска и конструкции.

Такие компоненты, как: невозвратного преобразования (перенаправления, сортировки,

канонизирующей перестановки («канонической разметки»), замены элементов) потока знаний, переключения, сравнения, слияния потоков знаний, можно отнести к компонентам деструкции потока знаний. К компонентам реконструкции потока знаний можно отнести компоненты: накопления и передачи потока знаний, перестановки элементов потока знаний и обмена элементов потока знаний. Компоненты: обратимого невозвратного преобразования (разветвления, дублирования) потока знаний, оптимизирующей трансляции (компиляции) потока поисковых запросов, относятся к компонентам конструкции потока знаний.

Оптимизирующая трансляция поисковых запросов является одной из основных задач, которая позволяет повысить быстродействие операций поиска, сокращая время решения соответствующей задачи. К основным этапам решения последней относятся: (1) построение оптимизированного разбиения образца поиска на граф потока данных поисковых операций и граф операций сравнения; (2) выделение множеств «аккумуляторов» для графа поиска – множеств вершин графа потока данных поисковых операций, которые находятся на одной орбите автоморфизма подграфа графа потока данных операций поиска; (3) построение линейного порядка вершин на графе потока данных операций поиска с «аккумуляторами», оптимизирующее количество итераций в процессе поиска; (4) преобразование графа операций поиска, адаптирующее его команды к системе команд целевого языка (группировка команд операций поиска ранее не найденного этой командой на текущей итерации элемента в дерево команд операций поиска множества элементов и выбора элемента из этого множества для формирования сочетаний из них и перестановок); (5) трансляция графа операций поиска в текст целевого языка.

Ещё одной важной задачей является задача накопления и передачи потока знаний, которая позволяет загрузить («наполнить») этапы массовой параллельно-распределённой системы ОПЗ. Для реализации соответствующих компонентов используются такие программные механизмы синхронизации, как события и мьютексы, основанные на механизмах синхронизации в активной графовой памяти и машине обработке онтологий, включая атомарные операции доступа к стеку машины обработки онтологий. Важными задачами являются также задачи сравнения и канонической разметки структур потока знаний. В общем случае, их решение, как и других задач, может быть реализовано множеством («коллективом») компонентов ОПЗ.

Для управления компонентами выделяют «управляющие» (становлением компонентов) компоненты, к которым относятся компоненты:

- деструкции этапов потока знаний;
- реконструкции этапов потока знаний;

– конструкции (этапов обратного распространения ошибок, панеобратимостных, су-возвратностных и других) этапов потока знаний,

включая компоненты обращения этапов потока знаний. Компоненты обратного распространения ошибок наряду с компонентами конструкции и реконструкции этапов обработки знаний рассматриваются как часть подсистемы обучения интеллектуальных агентов и систем.

II. АГЕНТЫ РЕШЕНИЯ ЗАДАЧ

Перечисленные компоненты, имеющие операции перехода от состояния к состоянию и обменивающиеся знаниями через общую активную графовую память, могут быть рассмотрены как агенты [2]. Среди агентов выделяют два основных класса [3]: обучаемые (интеллектуальные) агенты (ИА) и реактивные агенты (РА). К РА могут быть отнесены агенты: преобразования (замены элементов, слияния, конкатенации, дублирования) строк, преобразования (пересечения, объединения, суммы (симметрической разности)) множеств и sc-множеств, преобразования (обращения, композиции, становления элементов, становления множества, кондуктивной и индуктивной композиций и т.п.) метаопераций.

ИА (когнитивные компоненты) за счёт (семантического) протоколирования приобретают способности обучения, анализа протокола и построения гипотез, включая планирование.

ЗАКЛЮЧЕНИЕ

На базе ИП релизованы реактивные программные компоненты, также реализованы алгоритмы для прототипов компонентов оптимизированной трансляции поисковых запросов. Автор выражает благодарность Яльчику И. Н. за помощь в реализации прототипов программных компонентов ОПЗ и Иванову С. В. за помощь в реализации алгоритмов прототипов компонентов оптимизированной трансляции поисковых запросов.

1. Ивашенко, В. П. Модели решения задач в интеллектуальных системах. В 2 ч. Ч. 1 : Формальные модели обработки информации и параллельные модели решения задач : учеб. метод. пособие / В. П. Ивашенко. -- Минск : БГУИР, 2020. — 79 с.
2. Ivashenko, V. Application of an integration platform for ontological model-based problem solving using an unified semantic knowledge representation / V. Ivashenko // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2021) : сборник научных трудов / Белорусский государственный университет информатики и радиоэлектроники ; редкол. : В. В. Голенков [и др.]. – Минск, 2021. – Вып. 5. – С. 179–186.
3. Тарасов, В. Б. Агенты, многоагентные системы, виртуальные сообщества: стратегическое направление в информатике и искусственном интеллекте / В. Б. Тарасов // Новости искусственного интеллекта. – 1998. – №2. – С. 5–63

ТЕХНОЛОГИЯ РАЗРАБОТКИ ПРОГРАММНЫХ КОМПОНЕНТОВ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ НА ОСНОВЕ ИНТЕГРАЦИОННОЙ ПЛАТФОРМЫ

Ивашенко В. П.

Кафедра интеллектуальных информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: ivashenko@bsuir.by

Представлена технология программных компонентов интеллектуальных систем на основе интеграционной платформы, использующей унифицированное семантическое представление знаний и имеющей архитектуру системы, управляемой знаниями. Сформулированы требования интеграции моделей обработки знаний, соответствующих программным компонентам, выявлена общая типология программных компонентов, перечислены составляющие технологии, включая и поддерживающие программные компоненты, приведены свойства задач, решаемых на технологических этапах.

ВВЕДЕНИЕ

Интеграция в интеллектуальных системах (ИС) может быть рассмотрена в трёх аспектах: фронтальном – интеграция путём генерации новых ассоциаций между понятиями базы знаний, увеличивающих их плотность и обеспечивающих более полное представление («глубокое» «понимание») предмета предметной области; горизонтальном – путём добавления новых элементов (понятий) и моделей в базу знаний (БЗ), обеспечивающих более широкое представление, переход к предметным областям, расширяющим исходную предметную область; вертикальном – устранение «семантических разрывов» между разными уровнями представления и управления в ИС, обеспечивающее её способности к автономной работе и адаптации путём наблюдения, интроспекции и самообучения. Одним из существенных качеств интеграции является непрерывность процесса интеграции [1]. Технология разработки компонентов ИС, развивающая идеи непрерывной интеграции [1], предполагает приобретение новых знаний в процессе взаимодействия с внешней средой. Важно отметить, что интеллектуальные качества, исторически известные человеку, являются результатом существования их носителей в социуме. Таким образом, одной из сред приобретения знаний является социальная среда или коллектив интеллектуальных агентов. Как частный вид таких сред можно рассматривать интеллектуальные человеко-машинные системы, в которых люди и искусственные интеллектуальные агенты обмениваются сообщениями.

При взаимодействии агентов осуществляется решение задач исполнения и познания. Для того, чтобы обеспечить моделирование решения всех видов решений задач на основе метаоперации композиции, достаточно задать базис, который может состоять из задач поиска и деструкции, задач проверки или задач выбора.

Далее рассматриваются основы технологии, ориентированной на разработку компонентов ИС на основе интеграционной платформы (ИП) [2–3].

I. АРХИТЕКТУРА ИНТЕГРАЦИОННОЙ ПЛАТФОРМЫ

ИП использует клиент-серверную архитектуру взаимодействия машин обработки образов (МООбр) и онтологий и многоуровневую архитектуру систем, управляемых знаниями [2]. В соответствии с последней выделены уровни управления устройствами (УУУ), данными (УУД) и знаниями (УУЗ) [2]. На УУУ специфицируется процессор и память с линейным неограниченным адресным пространством для МООбр и активная графовая структурно-перестраиваемая память для машины обработки онтологий, которые обеспечивают базовую операционную семантику.

На УУД используются модель обработки строк и модель обобщённых формальных языков, формируемых в соответствии с переходом от операции замыкания Клини к операции с расширенной областью значений. Путём сведения семантики языков к теоретико-множественной, которая в свою очередь сведена к семантике ситуативных (растяжимых) множеств (с-множеств), и рассмотрения языков модели унифицированного семантического представления знаний (МУСПЗ), как аналогов симметрических ассоциативных графовых языков, обеспечивается интеграция представления данных за счёт равномогности, следующей из попарного взаимного погружения множеств языков модели, формальных языков и языков МУСПЗ [2].

На УУЗ используются модель спецификации знаний, МУСПЗ и модели обработки знаний (МОЗ) на основе МУСПЗ. В рамках МОЗ рассматривается операционная семантика, которая согласуется с денотационной семантикой МУСПЗ и сводится к понятиям актуального, неактуального и становления. Решение задач на УУЗ основано на их декомпозиции в соответ-

ствии с общей классификации задач (ОКЗ) [2] и интеграции моделей обработки знаний, имеющих унифицированное семантическое представление. Необходимыми требованиями интеграции моделей обработки знаний являются:

$$\begin{aligned} \forall \rho \exists i (\rho) (\rho = i \circ \pi \circ i (\rho) \circ \pi^{-1} \circ i^{-1}) \\ \forall \rho \exists i (\rho) (\pi^{-1} \circ i^{-1} \circ \rho \subseteq i (\rho) \circ \pi^{-1} \circ i^{-1}) \\ \forall \rho \exists i (\rho) (i (\rho) \subseteq \pi^{-1} \circ i^{-1} \circ \rho \circ i \circ \pi) \\ \forall \rho \exists i (\rho) (\rho \circ i \circ \pi \subseteq i \circ \pi \circ i (\rho)), \end{aligned}$$

где ρ – операция, i – биективное отношение интеграции, π – отображение текстового фрагмента в содержащий его текст (отношение включения (интеграции) фрагментов БЗ).

II. ПРОГРАММНЫЕ КОМПОНЕНТЫ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

Среди основных программных компонентов (БЗ) [3] в рамках технологии и ИП можно выделить: простые компоненты (индивидуальные); сложные компоненты (коллективные), интегрирующие другие компоненты. По признаку взаимодействия с МООбр выделяются: репрезентативные компоненты (РК), не отображающие своё состояние в МООбр; коммуникативные компоненты (КК), отображающие в МООбр своё состояние. В соответствии с ОКЗ и делением на задачи познания и исполнения выделяются: рецептивные компоненты (когнитивные компоненты); перформативные компоненты. Частным случаем рецептивных компонентов являются сенсорные компоненты, на состояние которых отображается состояние МООбр. К интерфейсно-диалоговым компонентам относятся сенсорные компоненты, которые являются коммуникативными.

В соответствии с ОКЗ и признаком обратимости выделяются: обратимостные компоненты; необратимостные компоненты (НК).

В соответствии же с признаками возвратности и безвозвратности выделяются: многократные компоненты (МК); полумногократные компоненты (ПМК); однократные компоненты (ОК). К постоянным компонентам относятся все МК и некоторые ПМК, к временным компонентам относятся некоторые ПМК и ОК. Циклическими компонентами являются перформативные обратимостные компоненты. Частным случаем циклических компонентов являются компоненты бытия. К демонстрационным компонентам относятся перформативные обратимостные КК. К поисковым компонентам относятся обратимостные РК, частным случаем поисковых КК являются навигационные компоненты.

III. СОСТАВЛЯЮЩИЕ ТЕХНОЛОГИИ

Основные технологические этапы разработки: (1) прототипных интерфейсно-диалоговых компонентов; (2) РК бытия; (3) демонстрационных компонентов; (4) обратимостных РК; (5)

навигационных компонентов; (6) необратимостных РК; (7) сенсорных и (необратимостных) КК. Особенностью технологии является чередование этапов разработки КК и РК. Структура смены этапов в общем случае нелинейная, допускается «возвращение» на предыдущие этапы. С целью недопущения утраты знаний и способности решать задачи технология в частности ориентирована на удовлетворение двух основных принципов: сувозвратности и панеобратимости. Сувозвратность сводится к возвратности частных компонентов – частей БЗ, но – при обратимости всей БЗ в целом. Панеобратимость сводится к погружению и интеграции НК, обладающих необратимым поведением, в обратимую БЗ. В основном технология направлена на сохранение свойства обратимости БЗ, которое, в свою очередь, является основой обеспечения принципа сувозвратности. Однако, технология не исключает вариантов нарушения свойства обратимости БЗ, например – в случае возвращения к начальному или более раннему состоянию БЗ, процесс же разработки для иных условий нарушения обратимости требует дополнительных исследований. Нарушение обратимости только для описанных вариантов и сохранение обратимости – в остальных будем называть «паобратимостью». Таким образом, задача разработки программных компонентов в рамках технологии может быть классифицирована как задача выбора.

В рамках технологии обеспечивается техническая поддержка этапов программными компонентами: семантического протоколирования [2], синтеза гипотез и планов по протоколу, интерактивных средств разработки БЗ [3], навигации, программирования и т.п.

ЗАКЛЮЧЕНИЕ

Проведена апробация компонентов ИП в соответствии с основными технологическими этапами при разработке компонентов справочно-проверяющих систем [2] и компонентов навигации по БЗ.

1. Duvall, Paul M. Continuous Integration. Improving Software Quality and Reducing Risk / Paul M. Duvall, S. M. Matyas, A. Glover. – Addison-Wesley, 2007. – 336 p.
2. Ivashenko, V. Application of an integration platform for ontological model-based problem solving using an unified semantic knowledge representation / V. Ivashenko // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2021) : сборник научных трудов / Белорусский государственный университет информатики и радиоэлектроники ; редкол. : В. В. Голенков [и др.]. – Минск, 2021. – Вып. 5. – С. 179–186.
3. Иващенко, В. П. Инструментальные средства разработки баз знаний / В. П. Иващенко // Дистанционное обучение – образовательная среда XXI века : материалы VI Междунар. науч.-метод. конф., Минск, 22–23 нояб. 2007 г. / Белорус. гос. ун-т информатики и радиоэлектроники. – Минск, 2007. – С. 182–185.

ПРИМЕНЕНИЕ ПОДХОДА К ПОСТРОЕНИЮ ПЛАНА ПОИСКА В БАЗЕ ЗНАНИЙ НА ОСНОВЕ СТОИМОСТИ ОПЕРАЦИЙ

Ивашенко В. П., Кальчевский Д. А.

Кафедра интеллектуальных информационных технологий,

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: ivashenko@bsuir.by

Рассматривается задача поиска по образцу в базе знаний и подход к её решению, основанный на методах теории расписаний и ориентированный на составление оптимизированного плана поиска. Для представления знаний используются однородные семантические сети на основе модели унифицированного семантического представления знаний.

ВВЕДЕНИЕ

В интеллектуальных системах, к которым относятся системы, основанные на знаниях, и системы, управляемые знаниями [1], одной из ключевых задач является задача поиска в базе знаний [2]. Это задача актуальна и для систем, использующих унифицированное семантическое представление знаний [1,2]. Задача поиска описывается с помощью поискового запроса, который константами описывает данные, известные в исходном состоянии, и описывает переменными данные, которые могут быть найдены или не найдены в каком-либо из целевых состояний. Запросы бывают простыми и сложными [3,4]. Простые состоят из одного образца поиска, а сложные – из нескольких. Образец поиска задаётся структурой семантической сети, обратное отношение отношению интерпретации которой является функцией. Далее рассмотрим задачу построения плана поиска для заданного образца поиска.

I. ЗАДАЧА ПОИСКА

Пусть задан образец поиска фрагмента семантической сети модели унифицированного семантического представления знаний:

- V – множество sc-элементов;
- E – множество связей отношения инцидентности sc-элементов ($E \subseteq V \times V$);
- T – матрица типов sc-элементов (размерность $|V| \times d$);
- M – матрица стоимостей операций поиска по связкам отношения инцидентности (размерность $d \times d \times s \times s$);
- W – матрица весов операций поиска по связкам отношения инцидентности (размерность $d \times d \times s$);
- μ – метрическая форма (размерность $s \times s$);
- c – мера стоимости.

Требуется построить оптимальный план поиска для одного исполнителя, где оптимизируемая функция – мера стоимости c .

Пусть $\mathbf{1}_S^V$ обозначает вектор значений индикаторной функции элементов S на V , который

имеет размер $|V|$. Определим функцию $c_i(p)$

$$c_i(p) = \sum_{j=1}^i \sum_{q=1}^j f(\langle p(j)_1, p(j)_2, p(q)_1, p(q)_2 \rangle),$$

где план поиска:

$$p \in \{r \mid r^{-1} \in \mathbf{N}^E\} \cap E^{\{n \mid n \leq |E|\}} \cap \mathbf{N},$$

а функция f :

$$f(\langle \chi, \gamma, \omega, v \rangle) = h(\langle \mathbf{1}_{\{\chi\}}^V * \mathbf{1}_{\{\gamma\}}^V, \mathbf{1}_{\{\omega\}}^V * \mathbf{1}_{\{v\}}^V \rangle)$$

Также пусть

$$h(\langle \beta^{mn}, \lambda^{rt} \rangle) = g(\lambda^{rt} * T_r^i * T_t^j * \beta^{mn} * W_{mn}^k)$$

и

$$g(\alpha^{ijk}) = (\alpha^{ijk} * M_{ijk}^p) * (\alpha^{ijk} * M_{ijk}^q) * \mu_{pq}$$

Тогда задача может быть сведена к задаче теории расписаний [5] $1|tree| \sum_{i=1}^n w_i * C_i$, для решения которой на каждом шаге достаточно проверить условие

$$f(\langle x, y, u, v \rangle) \leq f(\langle u, v, x, y \rangle)$$

или неотрицательность выражения

$$\Delta(\langle x, y, u, v \rangle) = f(\langle u, v, x, y \rangle) - f(\langle x, y, u, v \rangle)$$

Оптимизация ориентирована на то, чтобы как можно раньше выполнить большее количество проверок, что увеличит вероятность раннего завершения поиска в случае отсутствия очередных искомых данных при условии, что это отсутствие для любых искомых данных равновероятно. Также для запросов, образцы которых имеют циклическую структуру связей отношения инцидентности, оптимизация ориентирована на сокращение временных издержек поиска, связанных с разбиением образца поиска на ациклический образец поиска и граф проверок при условии, что операции проверки имеют стоимость заведомо меньшую стоимости любой операции поиска, и распределением направлений поиска для ациклического образца поиска, когда стоимости поиска в прямом и обратном направлении различны.

II. АЛГОРИТМ СОСТАВЛЕНИЯ ПЛАНА ПОИСКА

Для решения рассмотренной задачи рассмотрим следующий алгоритм, который получает на вход исходные данные, а на выходе возвращает два множества S и C .

1. Добавить все константы во множество $F(0)$ вершин с найденными значениями.
2. Множество рёбер $E(0) = E^{-1} \cup E$.
3. Множество $S(0) = \emptyset, t = 0$.
4. Для каждого ребра из $E(t) \cap (F(t) \times F(t))$ выполнить нижеописанную подпрограмму, перенести все рёбра из множества $E(t) \cap (F(t) \times F(t))$ во множество $C(t)$.
5. Проверить является ли непустым множество $M(t)$, если оно пустое, то перейти на пункт 9.

$$M(t) = (E(t)/C(t)) \cap (F(t) \times (V/F(t)))$$

6. Из множества $M(t)$ выбрать ребро $\langle x, y \rangle$ такое, что

$$0 \leq \min_{\langle u, v \rangle \in M(t)/\{\langle x, y \rangle\}} \{\Delta(\langle x, y, u, v \rangle)\}$$
7. В $F(t+1)$ добавить все вершины из $F(t)$ вместе с y , перенести во множество $S(t+1)$ ребра из $S(t)$ вместе с $\langle x, y \rangle$, ребро $\langle y, x \rangle$ исключить из $E(t)$ в $E(t+1)$.
8. Увеличить $t \leftarrow t+1$ и перейти на пункт 4.
9. Вернуть $\langle S(t), C(t) \rangle$.

Подпрограмма рассматривает две встречные ветви графа поиска $S(t)$ и заменяет в нём поиск значения вершины образца по направлению в первой ветви на поиск значения вершины образца во встречном направлении в ветви последней операции поиска, пока не выполняется некоторое условие, например:

$$w_i * C_{i-1} + \sum_{k=i}^z w_k * m_i \leq w_j * C_{i-1} + \sum_{k=j}^z w_k * m_j,$$

где i, j – номера операции поиска в прямом и во встречном направлениях ($z \geq j$), w – веса, m, C – времена и времена завершения операций. Исключённые дуги из графа поиска добавляются во множество $E(t)$, дуги в графе поиска и встречные им исключаются из множества $E(t)$, t увеличивается на соответствующее количество шагов. Возвращаются обновлённые $E(t), S(t), t$.

Например, есть образец поиска (Рис. 1), который может быть представлен графом [6]. В результате применения алгоритма можно получить следующий результат (Рис. 2) при условии задания некоторых значений T, M, W и μ .

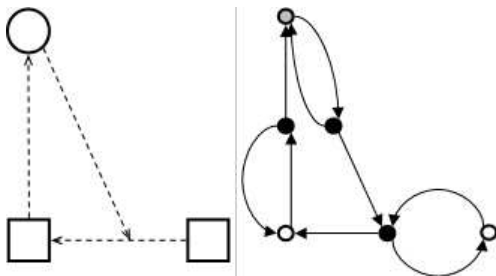


Рис. 1 – Образец поиска и графовое представление

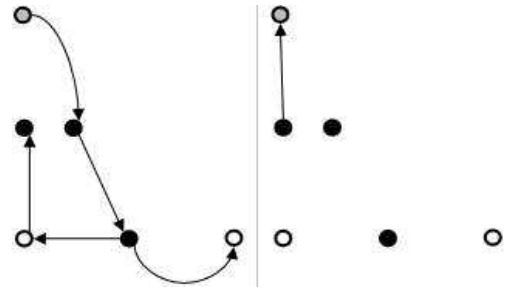


Рис. 2 – Дерево поиска и граф проверок (справа)

ЗАКЛЮЧЕНИЕ

Была проведена реализация разработанных алгоритмов средствами C/C++. Проведено тестирование и вычислительные эксперименты, подтверждающие работоспособность реализованных алгоритмов, разработан транслятор графов поиска на язык SCP [2]. Разработанные алгоритмы ориентированы на использование в рамках интеграционной платформы [1].

1. Ivashenko, V. Application of an integration platform for ontological model-based problem solving using an unified semantic knowledge representation / V. Ivashenko // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2021) : сборник научных трудов / Белорусский государственный университет информатики и радиоэлектроники ; редкол. : В. В. Голенков [и др.]. – Минск, 2021. – Вып. 5. – С. 179–186.
2. Программирование в графодинамических ассоциативных машинах / В. В. Голенков [и др.] – Минск : БГУИР, 2001, – 412 с.
3. Иващенко, В.П. Язык описания синтаксических правил для однородных семантических сетей / В.П. Иващенко // Дистанционное обучение – образовательная среда XXI века : материалы VI Междунар. науч. метод. конф., Минск, 22–23 нояб. 2007 г. / Белорус. гос. ун-т информатики и радиоэлектроники. – Минск, 2007. – С. 185–188.
4. Ivashenko, V. P. Application of optimization methods for search of semantic network constructions / V. P. Ivashenko // ECCO XVIII: Conference of the European chapter on combinatorial optimization: combinatorics for modern manufacturing, logistics and supply chains, Minsk, May 26–28, 2005 / Belarus. State Univ. ; ed. M.Y. Kovalyov. – Minsk, 2005. – P. 23–24.
5. Tanaev, V. S. Scheduling Theory: Single-Stage Systems / V. S. Tanaev, V. S. Gordon, Y. M. Shafransky. – Dordrecht; Boston; London: Kluwer Academic Publishers, 1994. – 380 p.
6. Иващенко, В. П. Представление семантических сетей и алгоритмы их организации и семантической обработки на вычислительных системах с массовым параллелизмом / В. П. Иващенко и другие // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2015) : материалы V междунар. науч. техн. конф. (Минск, 19–21 февраля 2015 года) / редкол. : В. В. Голенков (отв. ред.) [и др.]. – Минск : БГУИР, 2015. – С. 133–140.

ПОСТРОЕНИЕ ПРИКЛАДНОЙ ОНТОЛОГИИ СЕТЕЙ РАДИОСВЯЗИ

Карпук А. А., Говорко А. В.

Кафедра программного обеспечения сетей телекоммуникаций,

Белорусская государственная академия связи

Минск, Республика Беларусь

E-mail: A_Karpuk@mail.ru, govorko777@gmail.com

Предложен метод построения прикладной онтологии сетей радиосвязи. Даны формальные определения онтологии предметной области и онтологии задач сетей радиосвязи.

ВВЕДЕНИЕ

В работе [1] показано, что в современных информационных системах управления использованием радиочастотного спектра должны решаться новые задачи, для решения которых требуются глубокие знания о предметной области сетей радиосвязи. В базе знаний должны содержаться знания о технических характеристиках и координатах радиоэлектронных средств (РЭС), о технических характеристиках и составе радиолиний и радиосетей, о топографических и метеорологических характеристиках местности в районе размещения РЭС. Для хранения и извлечения этих знаний, а также для получения новых знаний, требуемых для решения задач управления сетями радиосвязи, предлагается построить прикладную онтологию сетей радиосвязи.

Прикладная онтология сетей радиосвязи строится на основе онтологии верхнего уровня, онтологии предметной области сетей радиосвязи и онтологии задач сетей радиосвязи. Онтологии верхнего уровня описывают общие понятия, такие как пространство, время, материя, объект, событие, действие и т.п., которые не зависят от конкретной задачи управления сетями радиосвязи. Онтология предметной области сетей радиосвязи описывает объекты сетей радиосвязи и отношения между ними. Онтология задач сетей радиосвязи определяет понятия и связи, описывающие методы извлечения знаний, требуемых для решения задач, из онтологии верхнего уровня и онтологии предметной области сетей радиосвязи, а также разбиение задач на подзадачи и программные компоненты, реализующие методы извлечения знаний и решение задач и подзадач.

I. ПОСТРОЕНИЕ ОНТОЛОГИИ ПРЕДМЕТНОЙ ОБЛАСТИ СЕТЕЙ РАДИОСВЯЗИ

В общем виде формальное определение каждой из перечисленных онтологий имеет вид четверки $O = \langle K, R, F, I \rangle$ [2], где K – конечное множество концептов (понятий) предметной области; R – конечное множество отношений между концептами; F – конечное множество функций интерпретации, заданных на концептах и отношениях; I – конечное множество аксиом, каждая из которых представляет собой всегда истин-

ное высказывание на концептах и отношениях. Для построения онтологии предметной области сетей радиосвязи представим множество концептов в виде $K = \langle D, A, Q \rangle$, где D – конечное множество доменов; A – конечное множество атрибутов; Q – конечное множество классов предметной области.

Домены используются как множества возможных значений атрибутов. Данные каждого домена имеют один из типов данных, допустимых в языке XML [3]. По количеству элементов данных в значении домена делятся на атомарные (atomic), объединенные (union) и списковые (list). Атомарный домен состоит из неделимых элементов данных определенного типа и формата. Значением объединенного домена является агрегат данных (структура), состоящая из других агрегатов и элементов данных. Значение любого объединенного домена можно представить в виде объединения значений входящих в него атомарных доменов. Значением спискового домена является список (повторяющаяся группа) значений атомарного или объединенного домена. Количество элементов списка может быть любым. Значение любого спискового домена можно представить в виде повторяющейся группы значений одного или нескольких атомарных доменов.

По ограничениям на значение атомарные домены делятся на примитивные (primitive), встроенные (built-in) и производные (constructed). В качестве примитивных атомарных доменов используются примитивные типы данных языка XML. Встроенные домены получают из примитивных доменов в результате применения к ним фиксированных ограничений на значения. Производные домены получают из примитивных и встроенных доменов в результате применения к ним различных ограничений на значения (fassets).

Концепты из множества A представляют собой свойства (атрибуты) классов предметной области. Каждый атрибут задается своим уникальным именем и доменом, которому принадлежат значения атрибута. В зависимости от того, на каком домене определен атрибут, он может быть атомарным, объединенным и списковым.

Концепты из множества Q представляют собой классы (объекты, сущности) предметной области. К одному классу относятся реальные или абстрактные люди, предметы, явления, события, процессы, имеющие одинаковый или близкий набор свойств (атрибутов), знания о которых хранятся в онтологии и используются при решении задач из данной предметной области. При построении онтологии предметной области сначала описываются классы, а затем в онтологию записываются знания об индивидах (individuals) каждого класса. В русскоязычной литературе индивиды классов часто называют экземплярами классов или объектов. Каждый класс задается своим уникальным именем и своим множеством атрибутов. Множество атрибутов класса, значения которых однозначно определяют индивид класса, объявляется ключом (key) класса. У класса может быть более одного ключа.

В множество отношений между концептами R входят отношения между доменами для построения производных доменов, отношения между атрибутами и доменами для определения области значения атрибутов, отношения между классами и атрибутами для определения состава атрибутов каждого класса, отношения между классами. Множество функций F состоит из n -арных отношений между классами или атрибутами, в которых значение элемента с номером n однозначно определяется значениями предыдущих $n-1$ элементов. Множество аксиом I служит для представления в онтологии высказываний о классах, атрибутах, доменах и отношениях, которые всегда истинны.

II. ПОСТРОЕНИЕ ОНТОЛОГИИ ЗАДАЧ СЕТЕЙ РАДИОСВЯЗИ

Для построения онтологии задач сетей радиосвязи представим множество концептов в виде $K = \langle T, O, Q, A, D, P \rangle$, где T – конечное множество задач; O – конечное множество онтологий предметной области или онтологий верхнего уровня (внешних онтологий), используемых при решении задач; Q – конечное множество классов из внешних онтологий или онтологий задач, используемых для получения входных данных и хранения выходных данных задач; A – конечное множество атрибутов из внешних онтологий или онтологий задач, используемых для получения входных данных и хранения выходных данных задач; D – конечное множество доменов из внешних онтологий или онтологий задач, используемых для представления входных и выходных данных задач; P – конечное множество программных компонентов, реализующих решение задач.

Множество T представим в виде объединения непересекающихся подмножеств $T = (AT \cup CS \cup CL)$, где AT – конечное множество ато-

марных задач (atomic Task); CS – конечное множество составных задач (composite Task); CL – конечное множество сложных задач (complex Task). К атомарным задачам относятся задачи, решаемые одним программным компонентом без обращения к другим программным компонентам. Решение составной задачи сводится к решению двух или более атомарных задач. В состав сложной задачи входят другие сложные задачи (подзадачи), составные и атомарные задачи. В качестве программных компонентов выступают функции библиотек динамической компоновки и исполняемые программные модули.

Концепты Q, A, D определяются аналогично определению этих концептов в онтологиях верхнего уровня и в онтологиях предметной области сетей радиосвязи. Отличие состоит в том, что в онтологию задач могут записываться индивиды этих концептов, являющиеся выходными данными задач. Индивиды этих компонентов, используемые в качестве входных данных задач и не являющиеся выходными данными других задач, выбираются из онтологий верхнего уровня и онтологий предметной области. Для этого концепты A и D онтологий задач связаны с концептом O этой онтологии отношением includedInOnto.

Входные данные каждой задачи из множества T описываются путем связывания концепта T с концептом A отношением hasInputAttr. Обратное отношение isInputAttrTask определяет вхождение атрибутов в состав входных данных задачи. Выходные данные каждой задачи из множества T описываются путем связывания концепта T с концептом A отношением hasOutputAttr. Обратное отношение isOutputAttrTask определяет вхождение атрибутов в состав выходных данных задачи. Программные компоненты для решения задач задаются путем связывания концепта T с концептом P отношением hasSoftComp. Вхождение атомарных задач в состав составных и сложных задач, а также вхождение составных задач в состав сложных задач описываются бинарным отношением на множестве T isSubTask. Обратное отношение hasSubTask определяет состав подзадач каждой сложной и составной задачи.

III. СПИСОК ЛИТЕРАТУРЫ

1. Карпук, А.А. Новые задачи управления использованием радиочастотного спектра и пути их решения / А.А. Карпук, А.В. Говорко // Проблемы инфокоммуникаций. – 2021. – № 1(13). – С. 56–63.
2. Палагин, А.В. Онтологические методы и средства обработки предметных знаний: монография / А.В. Палагин, С.Л. Крытый, Н.Г. Петренко. – Луганск: изд-во ВНУ им. В.Даля, 2012. – 324 с.
3. W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes. W3C Recommendation 5 April 2012 [Электронный ресурс]. – Режим доступа: <http://www.w3.org/TR/xmlschema11-2/>. – Дата доступа: 08.09.2021.

ПРИНЦИПЫ ПРЕДСТАВЛЕНИЯ СЕМАНТИЧЕСКИХ МОДЕЛЕЙ В ГРАФОВЫХ БАЗАХ ДАННЫХ

Лось П. В.

Кафедра интеллектуальных информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: p.los@bsuir.by

Анализируется и описывается возможный вариант представления семантических моделей в графовых базах данных с использованием таких технологий, как Neo4j и Apache Kafka.

ВВЕДЕНИЕ

Семантические модели в наши дни являются достаточно распространенной формой хранения и обработки информации в базах знаний интеллектуальных информационных систем. Семантическая паутина [1] и технология OSTIS [2] – лишь немногие примеры, где используется данный подход. Рассмотрим семантические модели немного подробнее.

I. АНАЛИЗ СЕМАНТИЧЕСКИХ МОДЕЛЕЙ И НАПРАВЛЕНИЙ ИХ ПРИМЕНЕНИЯ

Семантические модели, как понятно из названия, для представления информации (знаний) используют семантические сети – это ориентированный граф, вершины которого – понятия, а дуги – отношения между ними. В качестве понятий обычно выступают абстрактные или конкретные объекты, а отношения – это связи типа: «это», «имеет частью», «принадлежит» и др. Характерной особенностью семантических сетей является обязательное наличие трех типов отношений:

- класс – элемент класса;
- свойство – значение;
- пример элемента класса.

Семантическая паутина для представления таких графов использует URI – унифицированный идентификатор ресурса или адрес, используемый для указания ссылок на какой-либо объект. В основном вся информация представляется в виде текста, который больше подходит для восприятия людьми, но не для машинной обработки.

Для того, чтобы сделать их удобными для «машин», были предложены RDF, RDF Schema, OWL – язык описания семантической сети и два языка описания онтологий соответственно.

Иным примером технологии на основе семантических моделей является OSTIS. Она представляет собой технологию компонентного проектирования семантически совместимых интеллектуальных систем, содержащих базы знаний любого уровня сложности и реализующих параллельные модели обработки знаний. База знаний ostis-системы представляет собой семантическую сеть, имеющую сложную иерархическую струк-

туру, которая состоит из элементов, обозначающих не только внешние сущности и связи между ними, но и различные классы элементов семантической сети, связи между классами и фрагментами.

Языком представления знаний является SC-код, который позволяет описывать структуру любой информационной конструкции и хорошо приспособлен к использованию в условиях нестационарности и неточности, противоречивости, неактуальности, неполноты знаний. Существует несколько форм внешнего представления для SC-кода: SCn (natural), SCg (graphical), SCs (string) – они служат для представления знаний в удобной форме в зависимости от контекста.

II. ПРОБЛЕМЫ ХРАНЕНИЯ И ОБРАБОТКИ СЕМАНТИЧЕСКИХ МОДЕЛЕЙ НА СОВРЕМЕННЫХ КОМПЬЮТЕРАХ

Современные компьютеры не всегда хорошо подходят для интерпретации семантических моделей [3]. Это связано с низкой производительностью традиционных систем по причине последовательной обработки информации, низкого уровня доступа к памяти, её линейной организации и еще по ряду причин.

SC-код, который используется технологией OSTIS в качестве языка представления знаний, ориентирован на компьютеры нового поколения, в которых решены выше упомянутые проблемы, но таковые на данный момент не существуют. В связи с этим сейчас приходится моделировать их в виде программной реализации [4], что, конечно, отрицательно сказывается на производительности.

Таковой программной реализацией является sc-machine [5] – крайне эффективная имплементация обработки семантических сетей, написанная на языке программирования C++. Быстродействие достигается во многом за счет использования именно этого языка, так как он предоставляет программисту большой контроль над низкоуровневыми процессами, в том числе распределением и работой с памятью.

Существует большое количество агентов, которые и занимаются обработкой знаний, находящихся в базе.

Из-за некоторых моментов реализации у текущей версии есть ряд проблем, связанных с созданием масштабируемых, распределенных систем, поэтому предлагается другой подход к хранению и обработке этой информации.

III. ПРЕДЛАГАЕМЫЙ ПОДХОД

Как альтернативу предлагается использовать графовую систему управления базами данных (далее – СУБД) в качестве хранилища и событийно-ориентированный (event-driven) подход для обработки графов, а соответственно и знаний, которые представлены в виде семантических сетей.

Использование графовой СУБД уже дает преимущество во взаимодействии со знаниями, так как представление данных в базе, запросы в базу принимают «графовый вид», чем, по сути, и является семантическая сеть.

При этом одним из вариантов применения событийно-ориентированного подхода является создание серверных приложений в случае, когда нежелательно порождение лишних обслуживающих процессов. В многоагентной системе, к каковым относятся системы, разработанные на основе технологии OSTIS, происходит много вызовов различных агентов, они могут взаимодействовать друг с другом, что порождает дополнительные вызовы процедур. Таким образом, обработку всех этих вызовов необходимо вынести за рамки основного приложения, на что как раз таки в данном случае направлен событийно-ориентированный подход.

IV. ПРИНЦИПЫ РЕАЛИЗАЦИИ ПРЕДЛАГАЕМОГО ПОДХОДА

В качестве базы данных предлагается использовать Neo4j [6]. Neo4j – графовая СУБД с открытым исходным кодом, реализованная на Java. API для взаимодействия с Neo4j разработан для многих языков программирования, включая Python, Java, Go. Также программный интерфейс можно расширить с помощью серверных плагинов. К таковым, например, относится «Neo4j Streams Kafka Integration», который будет рассмотрен более подробно далее.

Для обмена сообщениями, событиями предлагается использовать Apache Kafka [7] – платформу распределенной потоковой передачи событий с открытым исходным кодом, которая используется тысячами компаний.

Принципиальное отличие Apache Kafka от других очередей задач и им подобным (AWS SQS, RabbitMQ) состоит в том, что сообщения (события) в очереди остаются после их обработки, что позволяет взаимодействовать с ними нескольким разным обработчикам. В отношении

многоагентных систем это отличие является довольно значительным в связи с тем, что одно и то же событие иногда должно быть обработано разными агентами.

Это же отличие позволяет обеспечить высокую степень отказоустойчивости и легкость восстановления после сбоев: Вам не придется после сбоя записывать все сообщения в очередь заново для их обработки, потому что они все еще находятся в очереди.

Как упоминалось ранее, для Neo4j есть плагин для интеграции с Apache Kafka – «Neo4j Streams Kafka Integration». Поток выполнения выглядит следующим образом:

1. основное приложение делает вызов удаленной процедуры с определенными параметрами;
2. обработчик видит этот вызов и начинает выполнять задачу;
3. по окончании выполнения обработчик сообщает о том, что результат готов и его необходимо «поднять» в основную базу;
4. основное приложение видит готовность задачи и обновляет данные в базе, используя результат выполнения задачи обработчиком.

V. ЗАКЛЮЧЕНИЕ

В работе рассмотрен подход к решению проблемы создания легко масштабируемых, распределенных систем на базе технологии OSTIS благодаря использованию графовых СУБД и событийно-ориентированного подхода.

1. W3C. Semantic Web – Mode of access: <https://www.w3.org/standards/semanticweb/>. – Date of access: 18.10.2021.
2. Голенков, В.В. Онтологическое проектирование гибридных семантически совместимых интеллектуальных систем на основе смыслового представления знаний / В.В. Голенков, Н.А. Гулякина, И.Т. Давыденко, Д.В. Шункевич, А.П. Еремеев // Онтология проектирования. – 2019. – Т. 9, № 31. – С. 132-151. – DOI: 10.18287/2223-9537-2019-9-1-132-151.
3. Семантические технологии проектирования интеллектуальных систем и семантические ассоциативные компьютеры / В.В. Голенков [и др.] // Доклады БГУ-ИП. – 2019. – № 3 (121). – С. 42-50.
4. Shunkevich, D. Ontological approach to the development of a software model of a semantic computer based on the traditional computer architecture / D. Shunkevich, D. Koronchik // Open Semantic Technologies for Intelligent Systems (OSTIS-2021): сборник научных трудов / Белорусский государственный университет информатики и радиоэлектроники; редкол.: В.В. Голенков [и др.]. – Минск, 2021. – Вып. 5. – С. 75-92.
5. sc-machine – Mode of access: <http://ostis-dev.github.io/sc-machine/>. – Date of access: 18.10.2021.
6. Neo4j – Mode of access: <https://neo4j.com/>. – Date of access: 18.10.2021.
7. Apache Kafka – Mode of access: <https://kafka.apache.org/>. – Date of access: 18.10.2021.

МОДЕЛИ, МЕТОДЫ И СРЕДСТВА ПОСТРОЕНИЯ ФОРМАЛЬНЫХ СЕМАНТИЧЕСКИХ МОДЕЛЕЙ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ

Новицкий В. А.

Кафедра интеллектуальных информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: novlad26@gmail.com

В данной работе рассматриваются принципы построения семантической модели мультимодальных пользовательских интерфейсов ostis-систем.

ВВЕДЕНИЕ

Пользовательский интерфейс представляет собой совокупность программных и аппаратных средств, обеспечивающих диалог пользователя и компьютера. Под диалогом в данном случае понимают процесс обмена информацией между пользователем и программой, осуществляемый в реальном масштабе времени и служащий для решения конкретной задачи [1].

Интерфейс пользователя – центральный элемент любой современной программной системы, поскольку системы существуют для того, чтобы с ними работали пользователи, а пользователи взаимодействуют с системой именно через интерфейс. Об эффективности интерфейса можно судить по тому, насколько эффективна работа пользователя с системой [2].

Причём, поскольку интеллектуальные системы имеют больше возможностей по сравнению с традиционными системами, то разработка их интерфейса становится довольно трудоёмкой задачей, а полученный в результате интерфейс является сложным для понимания пользователями. Для упрощения разработки и улучшения пользовательского опыта использования наиболее эффективным является использование формальных семантических моделей пользовательских интерфейсов интеллектуальных систем.

1. СЕМАНТИЧЕСКАЯ МОДЕЛЬ МУЛЬТИМОДАЛЬНЫХ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ OSTIS-СИСТЕМ

Проблемы унификации принципов построения различных компонентов компьютерных систем решаются в рамках Проекта OSTIS [3], направленного на создание открытой семантической технологии проектирования систем, управляемых знаниями. Системы, разрабатываемые по данной технологии, названы ostis-системами. В рамках данной статьи будем говорить о пользовательских интерфейсах ostis-систем.

В качестве формальной основы для построения моделей ostis-систем в рамках указанной технологии используется универсальный язык, названный SC-кодом. Тексты указанного языка записываются в виде однородных семантических

сетей с теоретико-множественной интерпретацией. Элементы таких семантических сетей называются sc-элементами, в частности, узлы – sc-узлами, связи – sc-коннекторами (sc-дугами, sc-ребрами).

В основе проектирования семантической модели пользовательских интерфейсов лежат следующие принципы:

- пользовательский интерфейс представляет собой специализированную ostis-систему, ориентированную на решение интерфейсных задач, и имеющую в составе базу знаний и машину обработки знаний пользовательского интерфейса, что даёт возможность пользователю адресовать пользовательскому интерфейсу различного рода вопросы;
- используется онтологический подход к проектированию пользовательского интерфейса, что способствует чёткому разделению деятельности различных разработчиков пользовательских интерфейсов, а также унификации принципов проектирования;
- используется SC-код в качестве формального языка внутреннего представления знаний (онтологий, предметных областей и др.), благодаря чему обеспечивается легкость интерпретации этих знаний и системой, и человеком – пользователем или разработчиком, а также однозначность восприятия этой информации ими;
- средствами SC-кода с помощью соответствующих онтологий описываются синтаксис и семантика всевозможных используемых внешних языков;
- трансляции с внутреннего языка на внешний и обратно организовываются так, чтобы механизмы трансляции не зависели от внешнего языка, для реализации нового специализированного языка в таком случае необходимо будет только описать его синтаксис и семантику, универсальная же модель трансляции не будет зависеть от данного описания;
- каждый элемент управления пользовательского интерфейса является внешним отоб-

ражением некоторого sc-элемента, хранящегося в семантической памяти (sc-памяти), что позволяет использовать их в качестве аргументов пользовательских команд и правильно трактовать прагматику и семантику объектов интерфейсной деятельности;

- предполагается выбор стилей визуализации, осуществляемый в зависимости от вида отображаемых знаний (например, использование различных элементов визуализации для одних видов знаний и других – для других), что позволит пользователю быстрее обучаться новым специализированным языкам, а также сделать простым и понятным отображение знаний;
- модель пользовательского интерфейса строится независимо от реализации платформы интерпретации такой модели, что позволяет легко переносить разработанную модель на разные платформы.

Как было сказано ранее, пользовательский интерфейс в рамках предлагаемого подхода является специализированной ostis-системой, а значит, необходимым его компонентом является база знаний [4]. База знаний пользовательского интерфейса включает в себя следующие части:

1. Описание процессов, относящихся к прошлому, настоящему и будущему пользовательского интерфейса. Под прошлым пользовательского интерфейса подразумеваются история его эксплуатации, а также эволюция интерфейса, под настоящим – текущее состояние пользовательского интерфейса, под будущим – планы развития пользовательского интерфейса. Анализ изложенных временных процессов позволяет делать оценки эффективности развития интерфейса и обеспечивает версию при проектировании пользовательских интерфейсов.
2. Модели пользователей, содержащие информацию об их особенностях, возможностях и предпочтениях, что в свою очередь позволяет интерфейсу быть гибким и адаптироваться к пользователю, обеспечивая максимально эффективное взаимодействие.
3. Типология действий пользователей и ostis-систем, позволяющая описать принципы организации взаимодействия пользовательского интерфейса с пользователями на всех уровнях интерфейсного взаимодействия.
4. Типология объектов этих действий, позволяющая произвести унификацию и согласование компонентов пользовательских интерфейсов, а также составить их иерархию.

5. Формальное описание внешних языков представления конструкций SC-кода, как универсальных, так и специализированных.

Если рассматривать Общую онтологию пользовательских интерфейсов, то имеет смысл говорить о системе четырёх взаимосвязанных онтологий:

- общая онтология пользовательских интерфейсов;
- общая онтология интерфейсных действий пользователей и ostis-систем;
- общая онтология действий разработчика пользовательского интерфейса;
- общая онтология инструментальных средств проектирования пользовательских интерфейсов.

Первая из них описывает общую модель пользовательских интерфейсов ostis-систем. Вторая – многоуровневую интерфейсную деятельность пользователя с одной стороны, и интерфейсную деятельность ostis-системы с другой стороны. В третьей онтологии рассматривается деятельность разработчика пользовательских интерфейсов ostis-системы. Наконец, четвёртая онтология рассматривает различные средства проектирования пользовательских интерфейсов, включая различные редакторы, трансляторы и прочий инструментарий.

ЗАКЛЮЧЕНИЕ

В рамках данной работы были рассмотрены принципы построения семантической модели мультимодальных пользовательских интерфейсов ostis-систем.

1. Гниденко, И. Г. Технология разработки программного обеспечения / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров – М.: Юрайт, 2019. – 235 с.
2. Белоусова, С. А. Анализ подходов к созданию пользовательского интерфейса / С. А. Белоусова, Ю. И. Рогозов // Известия ЮФУ. Технические науки. – 2014. – №6 (155). – С. 142–148.
3. Голенков, В. В. Семантическая технология компонентного проектирования систем, управляемых знаниями / В. В. Голенков, Н. А. Гулякина // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2015) : материалы V междунар. науч.-техн. конф., Минск, 19-21 февр. 2015 г. / Белорус. гос. ун-т информатики и радиоэлектроники ; редкол.: В. В. Голенков (отв. ред.) [и др.]. – Минск, 2015. – С. 57-78.
4. Зверуго, А. В. Описание онтологии предметной области интерфейсных действий пользователей / А. В. Зверуго, В. В. Трунц, А. Г. Шалёв // Информационные технологии и системы 2017 (ИТС 2017) = Information Technologies and Systems 2017 (ITS 2017) : материалы междунар. науч. конф., Минск, 25 окт. 2017 г. / Белорус. гос. ун-т информатики и радиоэлектроники ; редкол.: Л. Ю. Шилин [и др.]. – Минск, 2017. – С. 140-141.

ЭФФЕКТИВНЫЕ АЛГОРИТМЫ ДЕКОМПОЗИЦИИ ЛИНЕЙНЫХ НЕДООПРЕДЕЛЕННЫХ СИСТЕМ С МАТРИЦЕЙ ИНЦИДЕНТНОСТИ МУЛЬТИГРАФА

Пилипчук Л. А., Малявко А. В.

Кафедра компьютерных технологий и систем, Белорусский государственный университет
Минск, Республика Беларусь

E-mail: pilipchuk@bsu.by, maliaukas@gmail.com

Рассматриваются недоопределенные системы линейных алгебраических уравнений (СЛАУ) в которых наряду с уравнениями системы, составляющими вложенную сетевую структуру блочно-диагонального вида, содержится общая часть. При этом, каждый блок разреженной части системы является матрицей инцидентности графа. На основании результатов из теории сложности вычислений и теоретико-графовых свойств матрицы инцидентности графа получены эффективные методы декомпозиции линейных систем.

ВВЕДЕНИЕ

Исследование математического аппарата для решения разреженных СЛАУ с использованием вычислительных алгоритмов на графах является актуальной проблемой. Актуальность обусловлена необходимостью решать задачи разреженного матричного и сетевого анализа большой размерности на основе разделения переменных с использованием технологий алгоритмической теории графов.

I. СИСТЕМА БАЗИСНЫХ ЦИКЛОВ В МЕТОДАХ ПОСТРОЕНИЯ РЕШЕНИЙ РАЗРЕЖЕННЫХ НЕДООПРЕДЕЛЕННЫХ СЛАУ

Рассмотрим конечную ориентированную связную мультисеть $S = (I, U)$ без петель, где I — множество узлов, U — множество мультидуг, определенных на $I \times I$, $|I| < \infty$, $|U| < \infty$. Пусть K — множество различных типов потока в мультисети S , $|K| < \infty$. Обозначим через $S^k = (I^k, U^k)$ — связную сеть, соответствующую типу потока $k \in K$, где $I^k \subseteq I$ — множество узлов, U^k — множество дуг сети S^k для потока типа k соответственно. Определим следующие множества: $K(i): K(i) = \{k \in K : i \in I^k\}$ и $K(i, j) = \{k \in K : (i, j)^k \in U^k\}$ — множества типов потока, транспортируемых через каждый узел $i \in I$ и каждую дугу $(i, j) \in U$ соответственно. Введем подмножество мультидуг $U_0 \subseteq U$ такое, что для подмножества $K_0(i, j) \subseteq K(i, j)$ выполняется условие: $|K_0(i, j)| > 1$.

На введенной мультисети рассмотрим разреженную недоопределенную линейную систему следующего вида:

$$\sum_{j \in I_i^+(U^k)} x_{ij}^k - \sum_{j \in I_i^-(U^k)} x_{ji}^k = a_i^k, i \in I^k, k \in K, \quad (1)$$

$$\sum_{(i,j) \in U} \sum_{k \in K(i,j)} \lambda_{ij}^{kp} x_{ij}^k = \alpha_p, \quad p = \overline{1, q}, \quad (2)$$

$$\sum_{k \in K_0(i,j)} x_{ij}^k = z_{ij}, \quad (i, j) \in U_0, \quad (3)$$

где $I_i^+(U^k) = \{j \in I^k : (i, j)^k \in U^k\}$, $I_i^-(U^k) = \{j \in I^k : (j, i)^k \in U^k\}$; $a_i^k, \lambda_{ij}^{kp}, \alpha_p, z_{ij} \in \mathbf{R}$ — параметры системы; $x = (x_{ij}^k, (i, j)^k \in U^k, k \in K)$ — вектор неизвестных.

В работе [1] рассматриваются алгоритмы и технологии построения решений разреженных недоопределенных систем линейных алгебраических уравнений для однородного потока в сети S^k и их реализация в системе *Wolfram Mathematica*. В [1] также разработаны и применены структуры данных для построения базиса пространства решений однородной системы, порожденной из системы (1) (характеристические векторы $\{\delta^k(\tau, \rho), (\tau, \rho)^k \in U^k \setminus U_T^k\}$) для фиксированного $k \in K$. С учетом разреженности матрицы системы (1) и теоретико-графовых свойств корневых деревьев число операций алгоритма построения каждого вектора базиса пространства решений составляет $O(|I^k|)$ в наилучшем случае.

Теорема 1 Система порожденных характеристических векторов $\{\delta^k(\tau, \rho), (\tau, \rho)^k \in U^k \setminus U_T^k\}$ базисных циклов $\{C(\tau, \rho), (\tau, \rho) \in U^k \setminus U_T^k\}$ относительно заданного покрывающего дерева U_T^k образует базис пространства циклов в сети S^k . Другими словами, система порожденных характеристических векторов образует базис пространства решений однородной системы, порожденной из системы (1) для фиксированного $k \in K$ [2, 3].

Обозначим $\delta^k(C)$ — характеристический вектор произвольного цикла C в сети S^k со следующими компонентами: $\delta_{\tau\rho}^k(C) = 1$, если $(\tau, \rho)^k$ является прямой дугой цикла C , т.е. $(\tau, \rho)^k \in C^+ \subseteq C$; $\delta_{\tau\rho}^k(C) = -1$, если $(\tau, \rho)^k$ является обратной дугой цикла C , т.е. $(\tau, \rho)^k \in C^- \subseteq C$; $\delta_{\tau\rho}^k(C) = 0$, если $(\tau, \rho)^k \notin C$.

Теорема 2 Пусть задана система базисных циклов $\{C(\tau, \rho), (\tau, \rho) \in U^k \setminus U_T^k\}$ относительно некоторого покрывающего дерева U_T^k сети S^k

и пусть C – произвольный цикл в сети S^k , $k \in K$. Тогда характеристический вектор любого цикла C имеет представление:

$$\delta^k(C) = \sum_{(\tau, \rho) \in U^k \setminus U_T^k} \delta_{\tau\rho}^k(C) \delta^k(\tau, \rho).$$

На основании теорем 1 – 2 и декомпозиционного подхода [2, с. 15] система уравнений (1) – (3) представлена в виде $|K|$ независимых разреженных подсистем линейных алгебраических уравнений с матрицами инцидентности графов S^k , $k \in K$ и системы общего вида. Это позволяет находить решения разреженной недоопределенной системы с матрицей инцидентности графа S^k на надлежащих структурах данных для представления базисных циклов без использования обращения матриц.

II. ПРИМЕР ПОСТРОЕНИЯ РЕШЕНИЙ РАЗРЕЖЕННОЙ НЕДООПРЕДЕЛЕННОЙ СЛАУ

Рассмотрим пример построения решения разреженной недоопределенной системы линейных алгебраических уравнений для мультисети S , представленной на рисунке 1. Мультисеть представлена в виде объединения сетей $S^k = (I^k, U^k)$, $k \in K = \{1, 2, 3\}$, где множество K состоит из трех типов потока.

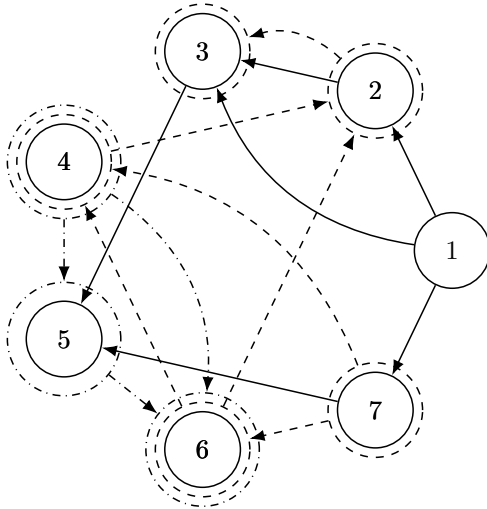


Рис. 1 – Объединение сетей $S^k = (I^k, U^k)$, $k \in K = \{1, 2, 3\}$

Выберем опоры U_T^k , $k \in K$, мультисети $S = (I, U)$ для системы (1) [2, с. 10]. Выбранная опора состоит из покрывающих деревьев U_T^k , $k \in K$. Сформируем множество циклических дуг $U_C = \{U_C^k \subseteq U^k \setminus U_T^k, k \in K\}$, $|U_C| = q + |U_0|$, где $U_C = \bigcup_{k=1}^3 U_C^k = \{(1, 3)^1, (5, 6)^2, (6, 2)^3\}$, $U_0 = \{(2, 3)\}$. Структуры, представляющие объединение множеств $U_T^k \cup U_C^k$, $k \in K$, представлены на рисунке 2. Дуги, входящие в U_T^k , $k \in K$, отображены более насыщенными линиями; циклические дуги U_C^k , $k \in K$ – менее насыщенными.

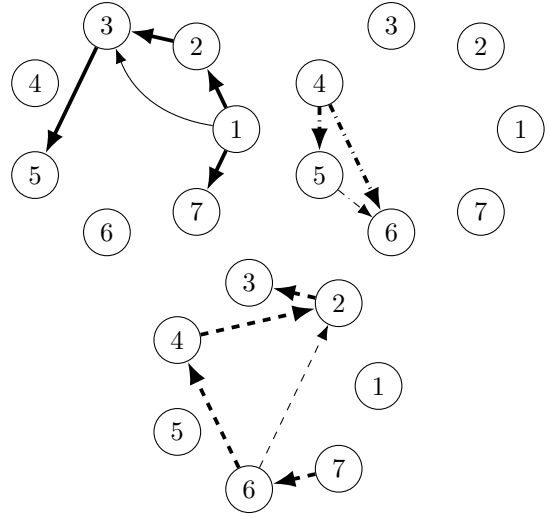


Рис. 2 – Множества $U_T^k \cup U_C^k$ для сетей S^k , $k \in K = \{1, 2, 3\}$

По правилам, указанным в [2], сформируем матрицу детерминантов $D = \begin{pmatrix} -11 & -7 & 10 \\ -5 & -1 & 0 \\ -1 & 0 & 0 \end{pmatrix}$, $\det D \neq 0$. Затем вычислим вектор правой части:

$$\beta = \begin{pmatrix} 88 + x_{7,5}^1 - 7x_{7,4}^2 \\ -5 - 3x_{7,5}^1 - 2x_{7,4}^2 \\ 6 + x_{7,5}^1 \end{pmatrix}, \quad x_C = \begin{pmatrix} x_{1,3}^1 \\ x_{5,6}^2 \\ x_{6,2}^3 \end{pmatrix}.$$

В случае невырожденной матрицы D дуговые потоки $x_C = (x_{\tau\rho}^k, (\tau, \rho)^k \in U_C^k, k \in K)$ вычислим однозначно из системы $x_C = D^{-1}\beta$.

С учетом разреженности матрицы системы (1) и теоретико-графовых свойств корневых деревьев [1] число операций алгоритма вычисления x_{ij}^k , $(i, j)^k \in U_T^k$ пропорционально числу вычисляемых элементов множества U_T^k , $\forall k \in K$.

ЗАКЛЮЧЕНИЕ

Разработаны эффективные алгоритмы декомпозиции линейных систем, которые основаны на результатах исследования структурных свойств фундаментальной системы циклов. Рассматривается пример решения разреженной недоопределенной системы линейных алгебраических уравнений для мультисети, состоящей из трех типов потока.

1. Пилипчук, Л. А. Алгоритмы и технологии построения решений разреженных недоопределенных линейных систем в Wolfram Mathematica / Л. А. Пилипчук, А. А. Лагуто // Технологии информатизации и управления (ТИМ 2016): материалы III Международной научно-практической конференции, Гродно, 14–15 апреля 2016 г. – Гродно: ГрГУ, 2016. – 9 с.
2. Пилипчук, Л. А. Разреженные недоопределенные системы линейных алгебраических уравнений / Л. А. Пилипчук. – Минск: БГУ, 2012. – 260 с.
3. Романовский, И. В. Дискретный анализ / И. В. Романовский – СПб.: Невский диалект; БХВ-Петербург, 2008. – 336 с.

СТАТИСТИЧЕСКИЕ МЕТОДЫ АНАЛИЗА ДАННЫХ ДЛЯ ОБНАРУЖЕНИЯ МОШЕННИЧЕСТВА

Сергей П. Д.

Кафедра программного обеспечения информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: polina.sergey.d@gmail.com

В данной работе рассматриваются статистические методы обнаружения мошенничества в банковской сфере, а именно неконтролируемые методы, также приводятся примеры некоторых известных способов анализа.

ВВЕДЕНИЕ

Статистические методы обнаружения мошенничества основаны на сравнении наблюдаемых данных и ожидаемых значений [1]. Цель статистического анализа заключается в том, чтобы в результате получить оценку подозрительности: чем выше оценка, тем больше вероятность того, что транзакция является мошеннической. Различают контролируемые и неконтролируемые методы обнаружения мошенничества. Контролируемые методы основаны обучении выявлению мошеннических и не мошеннических (законных) транзакций на обучающих наборах исходных данных, где заранее известно к какому классу - мошенническому или законному - принадлежит транзакция. Используя контролируемые методы, надо быть уверенным в достоверности обучающих данных. Проблемой для обучения является несбалансированность размеров классов, так как мошеннических транзакций значительно меньше законных. Вдобавок контролируемые методы можно использовать только для обнаружения тех типов мошенничеств, которые уже были замечены ранее. Неконтролируемые методы обнаружения мошенничества применяются при отсутствии обучающих наборов законных и мошеннических транзакций. Данные методы обычно являются сочетанием методов профилирования и отслеживания выбросов. Сначала моделируется базовое распределение, которое представляет нормальное поведение, а затем обнаруживается поведение, которое сильнее всего отклоняется от нормы.

I. АНАЛИЗ ОДНОРАНГОВЫХ ГРУПП (PEER GROUP ANALYSIS)

Метод анализа одноранговых групп отслеживает объекты, которые ведут себя иначе, чем объекты, с которыми у них раньше было схожее поведение [2]. Каждый объект рассматривается отдельно и сравнивается с остальными объектами с использованием критериев сравнения. На основе сравнения выбирается одноранговая группа, которая больше всего схожа с рассматриваемым объектом. Далее поведение изучаемого объекта сравнивается с поведением его одно-

ранговой группы, те объекты, поведение которые наиболее сильно выделяется на фоне поведения группы, получают большую оценку подозрительности. На рисунке 1 представлен пример выброса при использовании метода анализа одноранговых групп. Черной линией обозначена сумма денежных расходов рассматриваемого объекта за неделю, оранжевыми линиями - расходы одноранговой группы, голубыми - расходы остальных объектов. Расходы изучаемого объекта резко увеличились по сравнению с расходами объектов одноранговой группы, что говорит о возможном мошенничестве и требует дальнейшего изучения данного случая.

II. АНАЛИЗ ТОЧКИ ОСТАНОВА (BREAK POINT ANALYSIS)

Анализ точки останова – это метод обнаружения неконтролируемых выбросов на основе информации о транзакциях на одном банковском счете [3]. Точка останова – это наблюдение, которое считается аномальными для анализируемого счета. В анализе сравнивается последовательность транзакций, например, их сумма или частота в пределах движущегося окна транзакций фиксированной длины. Перед началом анализа необходимо установить длину окна и соотношение параметров старых и новых транзакций. Когда осуществляется новая транзакция, она попадает в окно, а самая старая транзакция в окне удаляется. Затем самые поздние транзакции из окна сравниваются с более ранними транзакциями из окна, чтобы понять, следуют ли они модели поведения ранних транзакций. На рисунке 2 показан пример точки останова. Черными вертикальными линиями обозначено окно наблюдений, транзакции до стрелки – транзакции из локальной модели, транзакции после стрелки отображают аномальное поведение для рассматриваемой модели. Преимуществом данного метода является то, что для него не требуются сбалансированные данные, так как рассматривается только транзакции одного счета и не требуется обобщать данные в фиксированные моменты времени, как в случае с методом анализа одноранговых групп.

ЗАКЛЮЧЕНИЕ

Использование исключительно контролируемых методов для обнаружения мошеннической транзакции может быть неэффективно, так как они предотвращают только известные схемы мошенничества. Для увеличения надежности системы предотвращения мошенничества целесообразно применять также неконтролируемые методы, которые могут отследить аномалии в поведении объектов и тем самым помочь выявить новые виды мошенничества.

СПИСОК ЛИТЕРАТУРЫ

1. Statistical fraud detection: a review / Bolton R. J, Hand D. J. // Statistical Science, vol. 17, no. 3, 2002, p. 235–255.
2. Peer Group Analysis / Bolton R.J. and Hand D.J. // Technical Report, Department of Mathematics, Imperial College, London, 2001, 15 p.
3. Unsupervised profiling methods for fraud detection / Bolton R.J. and Hand D.J. // Conference on Credit Scoring and Credit Control 7, Edinburgh, UK, 2001, 16 p.

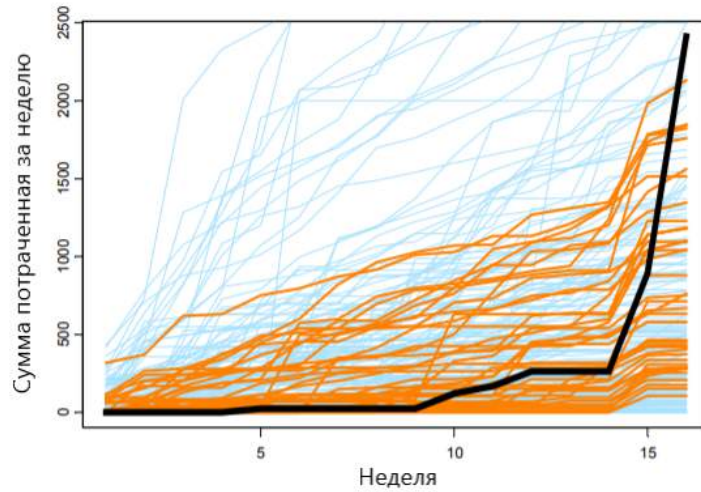


Рис. 1 – Пример выброса при использовании метода анализа одноранговых групп

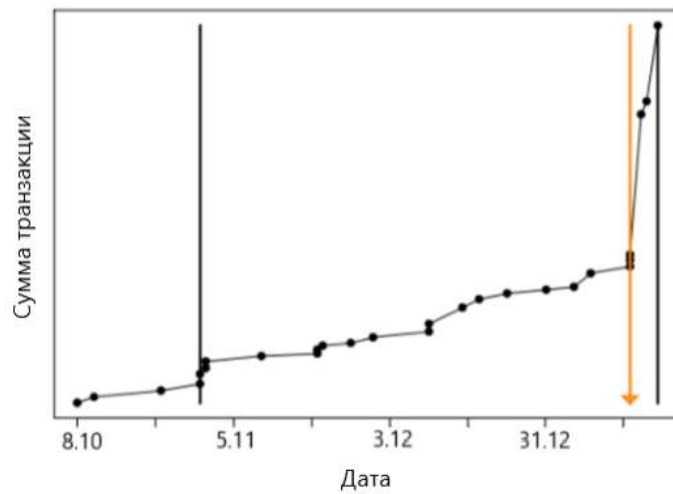


Рис. 2 – Пример точки останова

НЕЙРОСЕТЕВЫЕ МОДЕЛИ ДЛЯ БИНАРНОЙ КЛАССИФИКАЦИИ ДАННЫХ ТЕЛЕМЕТРИЧЕСКОЙ ИНФОРМАЦИИ МАЛЫХ КОСМИЧЕСКИХ АППАРАТОВ

Скобцов В. Ю.

Объединённый институт проблем информатики Национальной академии наук Беларуси
Кафедра ПОИТ, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: vasko_vasko@mail.ru

Разработаны и исследованы нейросетевые модели на основе современных архитектур глубокого обучения для решения задачи бинарной классификации данных телеметрической информации МКА, позволяющие определять штатное и нештатное состояние функционирования МКА или некоторых его подсистем. Проведены компьютерные эксперименты по обучению, валидации и тестированию построенных нейросетевых моделей, показавшие их достаточно высокую точность в решении рассматриваемой задачи.

ВВЕДЕНИЕ

Одной из важнейших задач на всех этапах жизненного цикла малых космических аппаратов (МКА) является анализ их телеметрической информации (ТМИ) о функционировании МКА с точки зрения определения их технического состояния для обеспечения их корректного и безопасного функционирования. Их актуальность обусловлена прежде всего тем, что одной из основных причин потерь МКА являются отказы, сбой и некорректная работа систем МКА.

Большое количество информации, поступающей и накапливающейся в специализированных банках данных с МКА, может быть эффективно использовано для определения технического состояния МКА и его подсистем. В современных условиях для решения задач обеспечения требуемой степени автономности, качества и оперативности управления такими сложными объектами как МКА необходимо выполнить комплексную автоматизацию и интеллектуализацию процессов многомодельного анализа данных ТМИ МКА. При этом в соответствии с ГОСТ 1410-002-2010 и Стратегией цифровой трансформации ракетно-космической отрасли до 2025 г. и перспективой до 2030г. госкорпорации «Роскосмос» важной задачей является создание так называемой системы информации о техническом состоянии и надежности космических комплексов и входящих в их состав изделий [1].

При этом разработка и применение методов анализа данных ТМИ МКА на основе моделей искусственного интеллекта и машинного обучения систем позволяет на новом научно-техническом уровне решить поставленную задачу и повысить эффективность принимаемых управленческих и эксплуатационных решений наземных комплексов управления (НКУ) МКА.

I. ЗАДАЧА БИНАРНОЙ КЛАССИФИКАЦИИ ДАННЫХ ТМИ МКА

Исходные данные ТМИ являются временным рядом, который можно представить как

матрицу, где i -я строка является анализируемым вектором показателей ТМИ в i -й момент времени, индекс j соответствует j -му показателю ТМИ в i -м векторе.

Определение 1. Одномерный временной ряд $\mathbf{X} = (x_1, x_2, \dots, x_T)$ – упорядоченный набор вещественных значений. Длина X равна количеству вещественных значений T .

Определение 2. M -мерный временной ряд $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_M)$ состоит из M различных одномерных временных рядов $\mathbf{X}_j \in R^T$.

Рассматриваемый временной ряд данных ТМИ является M -мерным временным рядом $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_M)$. Для каждого вектора показателей ТМИ в i -й момент времени \mathbf{X}_i в соответствие поставлена метка класса $y_i \in \mathbf{Y}$, который характеризует состояние функционирования анализируемого по данным ТМИ МКА или его подсистемы. В исследуемой задаче мы рассматриваем случай бинарной классификации, поскольку конечной целью является определить к штатному или нештатному состоянию принадлежит анализируемый вектор \mathbf{X}_i M -мерного временного ряда \mathbf{X} . В этом случае число классов $K = 2$ и, следовательно, $\mathbf{Y} \in \{0, 1\}$, где 0 обозначает штатное состояние и 1 – нештатное состояние анализируемой подсистемы МКА. Таким образом, стоит задача нахождения модели следующего отображения: $y : \mathbf{X} \rightarrow \mathbf{Y}$.

При этом будем использовать так называемое One Hot кодирование. В последнем случае вектору \mathbf{X}_i M -мерного временного ряда \mathbf{X} соответствует вектор $\mathbf{Y}_i = (y_{i0}, y_{i1})$ размерности $K = 2$. При этом в векторе \mathbf{Y}_i присутствует только одно значение 1, соответствующее метке класса 0 : (1, 0) или 1 : (0, 1).

II. АНАЛИЗ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ

В течение последних двух десятилетий классификация временных рядов считается одной из самых сложных проблем в области интеллектуального анализа данных [2,3]. Фактически, любая проблема классификации, использующая

данные, которые регистрируются с учетом некоторого понятия упорядочивания можно рассматривать как проблему классификации временного ряда. Временные ряды встречаются во многих реальных приложениях: электронных медицинских карт, распознавания человеческой деятельности, классификации акустических сцен, кибербезопасности, анализа состояний функционирования МКА по данным ТМИ [3,4].

Определение 3. Искусственной нейронной сетью (нейросетевой моделью) называется система, состоящая из совокупности связанных между собой по типу узлов направленного графа элементарных процессоров, называемых искусственными или формальными нейронами, и способная генерировать выходную информацию в ответ на входное воздействие.

Искусственный нейрон имитирует свойства биологического нейрона. Здесь множество входных сигналов, обозначенных $x_i, i = 1, \dots, n$, поступает на искусственный нейрон. Эти входные сигналы, в совокупности обозначаемые вектором $X = (x_1, x_2, \dots, x_n)$, соответствуют сигналам приходящим в синапсы биологического нейрона. Каждый синапс характеризуется величиной синаптической связи или ее весом w_i . Каждый входной сигнал умножается на соответствующий вес w_i , и поступает на суммирующий блок. Множество весов в совокупности обозначаются вектором W . Суммирующий блок, соответствующий телу биологического нейрона, складывает взвешенные входы алгебраически, создавая величину S . Полученная величина S подается на активационную выходную функцию нейрона a , моделируя процесс активации или нелинейную передаточную характеристику биологического нейрона. Таким образом, математическую модель искусственного нейрона можно представить выражением

$$y = a(S) = a\left(\sum_{i=1}^n w_i x_i + b\right),$$

где y - выходной сигнал нейрона, b - начальное возбуждение или смещение.

В анализе и разработке нейросетевых моделей мы будем рассматривать нейросетевые модели от простого к более сложному, начиная с основных на текущий момент нейросетевых моделей:

- полносвязных нейронных сетей/слоев (многослойных перцептронов, MLP);
- одномерных (1D) сверточных нейронных сетей/слоев (1D CNN);
- рекуррентных нейронных сетей/слоев типа Long Short-Term Memory (LSTM) и Gated Recurrent Units (GRU), и продолжая их комбинациями, в том числе, на основе архитектур ResNet и Inception [5].

В качестве активационных функций будем использовать следующие широко используемые активационные функции [6,7]:

- функция relu - rectified linear unit, линейный выпрямитель $\text{relu}(z) = \max(0, z)$;
- обобщение логистической функции для One-Hot кодирования меток классов $\text{softmax}(z_i) = e^{z_i} / \sum_{i=0}^1 e^{z_i}$.

В качестве функции ошибки будем использовать функцию двоичной кроссэнтропии, поскольку мы решаем задачу бинарной классификации [5].

Основываясь на описанных выше, нейросетевых архитектурах, были предложены и исследованы 3-слойная MLP модель, 5-слойная сверточная 1D CNN модель, 4-слойные рекуррентные LSTM и GRU модели. Каждая из них использовала финальный классификационный полносвязный слой из 2 нейронов с логистической функцией.

Компьютерные эксперименты проводились на реальных данных ТМИ одной из подсистем МКА. Каждый вектор матрицы ТМИ X_i имеет размерность 9 и помечается 0 в случае штатного состояния и 1 в случае нештатного состояния подсистемы. Общая размерность 9-мерного временного ряда X составляет 121690 векторов, из которых 77881 вектор составляет обучающий набор данных, 19471 вектор составляет валидационный набор данных, 24338 векторов составляет тестовый набор данных.

Для приведенной выше группы нейросетевых моделей обучение и валидация проводились при следующих значениях гиперпараметров: метод обучения adam (как один из наиболее эффективных на текущий момент), функция потерь - binary crossentropy, число эпох обучения - 500, размер мини-пакета - 128. Механизм раннего останова [5] не использовался и процесс обучения и валидации проходил на всех 500 эпохах.

Как показывают данные экспериментов, наименьшей точностью, как на этапе обучения - 0.8816, этапе валидации - 0.8810 и тестирования - 0.8809 обладает полносвязная модель, при этом время одной эпохи обучения и валидации является - 3 секунды. Более точной моделью (0.9065, 0.9037, 0.8999) является одномерная сверточная модель, причем время эпохи обучения и валидации модели такое же - 3 сек. Лидерами по точности, очевидно являются рекуррентные модели: модель GRU - (0.9485, 0.9358, 0.9336), модель LSTM - (0.9617, 0.9487, 0.9485). При этом по времени эпохи обучения и валидации модель GRU достаточно выигрывает у LSTM - 26 < 31 сек. Поэтому делаем вывод о лидерстве LSTM модели с точки зрения точности в данной серии экспериментов. Если же точность модели достаточно иметь больше 0.9 и важен фактор времени обучения и легковесности модели, то более привле-

кательной является сверточная модель 1D CNN. Увеличение числа слоев и нейронов в слоях моделей к увеличению качества моделей не привело, зачастую наблюдался обратный эффект выхода на плато и понижения точности в процессе обучения.

Дальнейшей целью исследований было с одной стороны повысить точность модели, с другой стороны уменьшить ее время обучения и валидации, то есть получить более легковесную модель по сравнению с рекуррентными. Для этого были последовательно соединены три блока слоев: сверточный 1D CNN, рекуррентный GRU или LSTM и в итоге как дискриминантный классификатор полносвязный блок:

```
Z = Conv1D(filters=512, kernel size=4,
activation='relu') Xi
Z = Conv1D(filters=512, kernel size=4,
activation='relu') (Z)
Z = Conv1D(filters=512, kernel size=4,
activation='relu') (Z)
Z = Pooling1D(2) (Z)
Z = Conv1D(filters=256, kernel size=2,
activation='relu') (Z)
Z = Conv1D(filters=256, kernel size=2,
activation='relu') (Z)
Z = Conv1D(filters=256, kernel size=2,
activation='relu') (Z)
Z = RNN(units=64) (Z)
Z = Dense(2, activation='softmax') (Z)
```

На основе данной архитектуры были получены несколько нейросетевых моделей путем использования в слое агрегации методов AveragePooling и MaxPooling, в рекуррентных слоях ячейки типа GRU и LSTM. Рассматривался входной временной ряд с исходными и нормированными значениями в диапазоне от 0 до 1, функцией MinMaxScaler. Обучение проводилось также на 500 эпохах, но при этом использовался механизм раннего останова в случае выхода на плато значения точности валидации в течение 10 итераций. Как показали эксперименты в этом случае обучение выполнялось не более 160 эпох. В рекуррентном слое применялась $l1$ и $l2$ регуляризация.

Как показали данные анализа, по точности и времени одной эпохи обучения и валидации в данной группе моделей с небольшим преимуществом лидирует модель с параметрами AveragePooling, GRU и показателями точности на обучающем, валидационном и тестовом этапах: 0.9850, 0.9668, 0.9661, и временем одной эпохи обучения и валидации 14 сек.

Следующая группа моделей строилась на основе архитектуры предыдущей модели и методики добавления остаточных связей, развитие которой началось с появления семейства сетей ResNet [5]. Также было увеличено число сверточных слоев и уменьшено число фильтров в них.

```
Z1 = Conv1D(filters=64, kernel size=4,
```

```
activation='relu') (Xi)
```

```
Z1 = Conv1D(filters=64, kernel size=4,
activation='relu') (Z1) * 9 слоев
```

```
Z2 = add([Z1, Xi]) – пропуск остаточной связи
входа Xi
```

```
Z2 = Pooling1D(2) (Z2)
```

```
Z2 = Conv1D(filters=64, kernel size=2,
activation='relu') (Z2) * 9 слоев
```

```
Z3 = Conv1D(filters=64, kernel size=2,
activation='relu') (Z2)
```

```
Z4 = Pooling1D(2) (Xi)
```

```
Output = add([Z2, Z3, Z4]) – пропуск остаточных
связей Z2 и Z3
```

```
Output = Dense(32, activation='relu') (Output)
```

```
Output = Dense(2, activation='softmax') (Output)
```

По данным экспериментов, исходя из соотношения значения точности и функции потерь, на этапах обучения, валидации и тестирования, а также времени одной эпохи обучения и валидации лидером является модель с параметрами: AveragePooling, GRU, нормированные данные с показателями точности на обучающем, валидационном и тестовом этапах: 0.9821, 0.9665, 0.9690, и временем обучения эпохи 12 сек.

Также исследовалась группа моделей на основе архитектуры семейства Inception [5]. К сожалению, прироста в значениях точности и уменьшения значений потерь не произошло. При этом время одной эпохи обучения и валидации возросло более чем в 2 раза.

Машинный анализ разработанных моделей показал преимущество нейросетевых глубоких моделей, представляющих собой последовательное соединение трех блоков слоев: сверточного 1D CNN, рекуррентного GRU и итогового полносвязного блока- классификатора, с применением слоя агрегации AveragePooling, метода пропуска остаточных связей семейства архитектур ResNet и нормирования исходного набора данных в силу небольшого преимущества по точности на этапе тестирования и времени обучения и валидации. Также хорошо себя показала подобная модель без остаточных данных и нормирования входных данных, но с немного более длительным временем эпохи обучения и валидации.

1. ГОСТ РО 1410-002-2010. Ракетно-космическая техника. Система информации о техническом состоянии и надежности космических комплексов и входящих в их состав изделий.
2. Yang Q. 10 challenging problems in data mining research /Yang Q., Wu X. // Inf. Technol. Decis. Mak. – 2006. – 05(04). – pp.597–604.
3. Hassan Ismail Fawaz, Deep learning for time series classification: a review /Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, Pierre-Alain Muller // Data Mining and Knowledge Discovery. – 2019. – 33. – pp.917–963.
4. Скобцов В. Ю. Нейросетевой анализ данных телеметрической информации бортовой аппаратуры космических аппаратов. /Скобцов В. Ю., Архипов В. И. // Космическая техника и технологии. – 2021. – С.111-124.
5. Шолле Ф. Глубокое обучение на Python / Шолле Ф. // Издательство: МЦНМО, 2006. – 448 с.

ПРИМЕНЕНИЕ АЛГОРИТМОВ ДЛЯ ОПРЕДЕЛЕНИЯ ПЛАГИАТА В ПРОГРАММНОМ КОДЕ

Удовин И. А., Воронова В. В.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: wilcot@ya.ru, veronika.voronova31@gmail.com

В данной статье рассматриваются алгоритмы применяемые для определения плагиата в программном коде. Проанализированы основные проблемы связанные со сложностью простого сравнения исходного кода программ, а также алгоритмы, позволяющие избежать данные проблемы. Была рассмотрена поисковая система, которая позволяет масштабировать обнаружение плагиата на больших объемах программного кода.

ВВЕДЕНИЕ

Плагиат является довольно распространенной проблемой в оцениваемых заданиях. На курсах по программированию студенты часто прибегают к плагиату исходных кодов программ, доступных в открытом доступе. Плагиат программного кода встречается и на соревнованиях по спортивному программированию, где количество проверяемых программ может достигать нескольких тысяч или десятков тысяч. Ручное определение плагиата в таких случаях нецелесообразно, если вообще осуществимо, за разумные сроки. Поэтому для определения плагиата необходимо использовать автоматизированные алгоритмы, позволяющие обрабатывать большие объемы исходных кодов программ.

Плагиат – это выдача чужого произведения за своё или незаконное опубликование чужого произведения под своим именем. Касательно программного кода, это относится к случаям копирования чужого кода. Причем в одних случаях копирование может осуществляться путем некоторого изменения названий объектов, изменением порядка кода, а в более сложных случаях – изменением структуры и замены одних конструкций другими. Также бывают случаи плагиата, когда программный код переписывается с одного языка программирования на другой. Если в случае простого копирования чужого кода можно обойтись простыми алгоритмами, например поиска наибольшей общей подстроки, то в остальных случаях необходимы более сложные алгоритмы, в том числе использующие методы машинного обучения.

Есть два фактора, которые усложняют обнаружение плагиата — это обилие доступных ресурсов и разнообразие методов, используемых для маскировки скопированных материалов. Есть ряд подходов для обнаружения плагиата в исходном коде программы, некоторые из которых будут рассмотрены в этой статье.

I. СРАВНЕНИЕ ФАКТОРОВ

При сравнении факторов, сходство двух программ оценивается по сходству различных

показателей программного обеспечения, таких как среднее количество символов в строке, количество строк комментариев, количество строк с отступами, количество пустых строк и количество маркеров (например, ключевые слова, символы оператора и имена модулей стандартной библиотеки). Один из подходов сравнивает две программы на основе трех профилей:

- Физический профиль характеризует программу на основе ее физических атрибутов, таких как количество строк, слов и символов;
- Профиль Холстеда характеризует программу на основе использования типов токенов и их частот. Они включают количество вхождений токенов (N), количество уникальных токенов (n) и их объем ($N \log_2 n$);
- Составной профиль комбинация физического профиля и профиля Холстеда.

Для выявления плагиата профили каждой программы вычисляются, а затем нормализуются. Сходство двух программ оценивается путем вычисления евклидова расстояния между их профилями. Системы, которые используют сравнение факторов, могут быть очень нечувствительными (могут быть легко введены в заблуждение) или очень чувствительными (приводящими к множеству ложных срабатываний), поскольку они игнорируют структуру программы. Нарушители могут легко добавлять или удалять комментарии, переменные или избыточные строки кода, чтобы избежать обнаружения плагиата [1].

II. СРАВНЕНИЕ СТРУКТУРЫ

Этот подход основан на убеждении, что сходство двух программ можно оценить по сходству их структур.

При таком подходе программы сравниваются в два этапа. Первый этап анализирует код и генерирует последовательности токенов, в то время как второй этап сравнивает токены.

Один из таких подходов – Sim. Он обнаруживает сходство между программами, оценивая их правильность, стиль и уникальность. Каждая программа сначала анализируется с помо-

щью лексического анализатора, который генерирует последовательность токенов. Токены для ключевых слов, специальных символов и комментариев предопределены, в то время как токены для идентификаторов назначаются динамически и хранятся в общей таблице символов, пробелы при этом отбрасываются. Поток токенов второй программы сгруппирован в разделы, каждый из которых представляет модуль программы. Каждый раздел отдельно выровнен с потоком токенов первой программы. Выравнивание двух строк выполняется путем вставки пробелов между символами для выравнивания их длины. Для расчета сходства используется схема оценки выравнивания. [2]

III. СРАВНЕНИЕ ПРОМЕЖУТОЧНОГО КОДА

Существующие системы обнаружения плагиата, основанные на сравнении факторов или структур, разработаны для обнаружения плагиата на определенном языке, таком как C, Java, Pascal и т.д. Мы называем это внутриязыковым плагиатом. Однако их такие подходы сложно применить для выявления межъязыкового плагиата, когда код на одном языке является плагиатом и отображается на другом. Для определения межъязыкового плагиата методом сравнения промежуточного кода необходимо разработать компилятор, способный генерировать промежуточный код для нескольких языков программирования.

Генерация промежуточного кода обычно происходит в несколько этапов:

1. Лексический анализ;
2. Синтаксический анализ;
3. Семантический анализ.

IV. ЯЗЫК ПЕРЕДАЧИ РЕГИСТРОВ

Язык передачи регистров (RTL) GCC содержит ряд инструкций, представленных во вложенных круглых скобках. Каждая инструкция содержит номер строки, указатель на предыдущую инструкцию и указатель на следующую инструкцию, за которой следуют выражения. GCC обеспечивает три уровня оптимизации компилятора.

В оптимизированном RTL инициализация переменных выполняется путем сохранения значения в виртуальном регистре (представленном маркером `reg`), а не в виртуальном стеке (представленном `reg` и значением смещения). Это делает RTL-файл и, следовательно, индекс поисковой системы более компактным. Выбор оптимизации также приводит к вставке функций (инлайнинг), когда компилятор вставляет функции

короче порогового значения (по умолчанию 600 строк) в вызывающий функцию код. На практике получается, что самый высокий уровень оптимизации приносит наибольшую пользу такому подходу, поскольку инструкции RTL максимально упрощены. Кроме того, оптимизация позволяет легко обнаруживать случаи, когда вставка функций используется для маскировки скопированного кода. GCC также предоставляет возможность компиляции для добавления отладочной информации в промежуточный код, однако некоторые эксперименты показывают, что эта дополнительная информация бесполезна.

V. ПОИСКОВАЯ СИСТЕМА

Система обнаружения плагиата, основанная на попарных сравнениях, плохо масштабируется. Хорошим альтернативным подходом является индексирование и запрос токенов с помощью поисковой системы. [3]

В таком случае, поисковая система может использоваться для выполнения двух задач:

1. На этапе индексирования индексируется сгруппированный RTL файлов исходного кода в коллекции;
2. На этапе обнаружения сгруппированный RTL исходного кода запроса используется для поиска по индексу, и программы из коллекции перечисляются в порядке убывания сходства с исходным запросом.

VI. ЗАКЛЮЧЕНИЕ

В данной статье были рассмотрены некоторые алгоритмы обнаружения плагиата в программном коде. Были рассмотрены методы основанные на сравнении факторов, сравнении структуры программ, а также методы основанные на сравнении промежуточного кода и RTL. Была также рассмотрена проблема масштабирования алгоритмов обнаружения плагиата в программном коде путем использования поисковой системы.

СПИСОК ЛИТЕРАТУРЫ

1. Arwin, C., Tahaghoghi, S. M. M. Plagiarism detection across programming languages. Proc. 29th Australasian Computer Science Conf.. – 2006. – Vol. 48. – P. 277–286.
2. Gitchell, D., Tran, N. Sim: a utility for detecting similarity in computer programs, in 'Proceedings of the Thirtieth SIGCSE Technical Symposium on Computer Science Education', ACM Press. – 1999. – P. 266–270.
3. Burrows, S., Tahaghoghi, S. M. M., Zobel, J. Efficient and effective plagiarism detection for large code repositories, in 'G. Abraham and B.I.P. Rubinstein Editors, Proceedings of the Second Australian Undergraduate Students' Computing Conference (AUSCC04)'. – 2004. – P.8-15.

СИСТЕМА ОЦЕНКИ ЗНАНИЙ УЧАЩИХСЯ С ПРОГНОЗИРОВАНИЕМ УСПЕВАЕМОСТИ

Халецкий А. М.

Кафедра интеллектуальных информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail:

Данная работа посвящена системам, которые учитывают результаты обучающегося, а так же системам, которые описываются как набор инструментов и методик, которые используются для предоставления предложений обучающемуся. Описаны проблемы рекомендаций, которые показывают, что рекомендация, как правильно определятся не на всем пространстве, а только на каком-то его подмножестве. Описана оценка системы, а так же типы систем фильтрации рекомендаций, которая объясняет, что системы учатся рекомендовать элементы, которые схожи с теми, которые понравились данному пользователю в прошлом.

ВВЕДЕНИЕ

Каждый обучающийся стремится выбрать курсы, которые отвечают его интересам и возможностям. Это решение может повлиять на мотивацию и его будущие достижения. Educational Data Mining (EDM) описывает область исследований, которая связана с разработкой методов, использующих несколько типов данных. Несмотря на то, что были достигнуты результаты по моделированию и прогнозам успеваемости, когда речь идет о рекомендациях по курсам, мы не можем найти большего разнообразия. Тем не менее, результаты, которые дают системы на других направлениях, признаны, но проблема их применения в данном случае мало изучена. Современные системы рекомендуют курсы в связи с их содержанием, не учитывая того, как этот курс влияет на успеваемость обучающегося. Поэтому появилось предложение создать систему, которая, учитывая результаты обучающегося, автоматически пересматривает потенциально интересные обучающемуся курсы, более подходящие его текущим навыкам.[1]

I. РЕКОМЕНДАТЕЛЬНЫЕ СИСТЕМЫ

Системы рекомендаций стали важной междисциплинарной областью исследований с середины 1990-х годов. Было посвящено большое количество времени и усилий по решению проблем данной области.

Систему рекомендаций можно описать как набор инструментов и методик, используемых с целью предоставления предложений по различным объектам лицам, не обладающим достаточной компетенцией для оценки потенциально огромного числа имеющихся альтернатив. Рекомендации стали использоваться в электронной торговле, в сфере доступа к информации, развлечений и различных видов услуг, предоставляя предложения, которые эффективно сокращают большие информационные пространства, чтобы пользователи были направлены к тем событиям

и услугам, которые наилучшим образом отвечают их потребностям и предпочтениям. [1–2]

II. ПРОБЛЕМЫ РЕКОМЕНДАЦИЙ

Основная проблема рекомендательных систем заключается в том, что рекомендация, как правило, определяется не на всем пространстве, а только на каком-то его подмножестве. Это означает, что необходимо экстраполировать от известных до неизвестных оценок, либо с помощью эвристики, либо оценочных функций.

При проектировании систем рекомендаций необходимо учитывать некоторые свойства, чтобы оценить, какое из них является наилучшим решением проблемы. Одним из заметных свойств большинства систем рекомендаций является то, что они должны иметь возможность получать обратную связь от действий пользователей, как явно, так и неявно. Примеры явных данных включают рейтинг элементов, ранжирование элементов и задание вопросов пользователю. Примеры неявных данных включают анализ социальных отношений, просмотров предметов и конкретного поведения пользователя. Такой поиск обратной связи необходим для построения точного профиля пользователя, однако, если система слишком навязчива, пользователям это может не понравиться.[3]

III. ОЦЕНКА СИСТЕМ РЕКОМЕНДАЦИЙ

Оценка систем рекомендаций не является тривиальной задачей, поскольку качество выдаваемых рекомендаций неразрывно связано с используемыми алгоритмами (их производительность может меняться с изменением размера данных) и с основной целью системы. Таким образом, решение вопроса о том, какую комбинацию критериев использовать, является существенной задачей, которая должна быть решена для каждой отдельной системы рекомендаций и сравнительной оценки. Подсчеты могут быть завершены как с использованием автономного

анализа, быстрого и экономически обоснованного прогнозирования определенных значений из набора данных, так и с учетом реального опыта пользователя. [4]

Системы рекомендаций предлагают три класса метрик точности рекомендательной системы:

1. Метрики прогностической точности измеряют, насколько предсказанные оценки системы рекомендаций близки к истинным оценкам пользователей;
2. Метрики точности классификации измеряют, сколько раз система имеет правильное суждение о том, является ли элемент хорошим для пользователя или нет;
3. Метрики ранговой точности измеряют способность алгоритма составлять упорядоченный список элементов, который соответствует потребностям пользователя.

IV. ТИПЫ СИСТЕМ ФИЛЬТРАЦИИ РЕКОМЕНДАЦИЙ

С использованием контентной фильтрации система учится рекомендовать элементы, которые схожи с теми, которые понравились данному пользователю в прошлом. Таким образом, с использованием контентной фильтрации достигается независимость пользователя и быстро генерируются рекомендации для нового элемента в базе данных, так как рекомендации основываются только на наборе характеристик элемента. Пример контентной фильтрации рекомендаций представлен на (Рис.1)

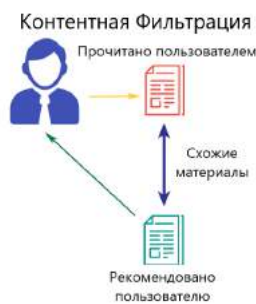


Рис. 1 – Пример контентной фильтрации рекомендаций

В коллаборативной фильтрации рекомендаций проходит идентификация пользователей, вкусы которых схожи со вкусами данного поль-

зователя, и рекомендовать понравившиеся им элементы, никогда не проводя никакого анализа по этим элементам вообще. Пример коллаборативной фильтрации рекомендаций представлен на (Рис.2)[5–6]



Рис. 2 – Пример коллаборативной фильтрации рекомендаций

V. ЗАКЛЮЧЕНИЕ

Студентам все еще сложно выбрать для себя оптимальный набор курсов, и в большинстве случаев у них нет необходимой поддержки и консультаций, необходимых для принятия решения. В данной публикации мы показываем, что можем использовать известные методы рекомендаций, чтобы рекомендовать студентам курсы, которые не только интересны им, но и наиболее схожи с их навыками.

VI. СПИСОК ЛИТЕРАТУРЫ

1. Messina, E. R. Measuring the Performance and Intelligence of Systems / E. R. Messina, A. M. Meystel // N. Engl. . – 2001. – Vol. 342, № 982. – P. 164.
2. Системы искусственного интеллекта. Учебное пособие. / М. В. Бураков . – СПб.: Инкарт, 2010. – 440 с.
3. Флах, П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных / пер. с англ.// Вест. аритмол. – 2015. – С. 400.
4. Самообучающиеся системы. / С. И. Николенко, А. Л. Тулупьев // – МЦНМО, 2009. – 288 с.
5. Vialardi, C. Recommendation in higher education using data mining techniques / C. Vialardi, J. Bravo // N. Engl. – May 2009., – P. 190–199/
6. Gutierrez, F. Providing grades and feedback for student summaries by ontology-based information extraction / F. Gutierrez, D. Dou // N. Engl. – September 2012., – P. 1722–1726/

AN APPROACH TO CALCULATING THE SIMILARITY BETWEEN SEMANTIC SEGMENTS IN THE INTELLIGENT TUTORING SYSTEMS

Li Wenzu, Qian Longwei.

Department of intellectual information technologies,
Belarussian state university informatics and radioelectronics
Minsk, Belarus

E-mail: lwzzggml@gmail.com, qianlw1226@gmail.com

This article proposes an approach to develop a sc-agent for calculating the similarity between semantic fragments described based on factual knowledge in the intelligent tutoring systems.

INTRODUCTION

With the rapid development of artificial intelligence technology, many intelligent tutoring systems (ITS) based on artificial intelligence technology have been developed. One of the most basic functions of ITS is to automatically verify the correctness and completeness of user answers. In the ITS, according to the type of knowledge describing the answer to the question, answer verification can be divided into: 1. answer verification of factual knowledge type (factual knowledge does not contain variables represents facts); 2. answer verification of logical knowledge type (logical knowledge contains variables, used to describe logical expressions). The most critical step in answer verification in ITS developed using OSTIS Technology (ostis-systems) is to calculate the similarity between the semantic fragment of the standard answer and the semantic fragment of the user answer. Due to the limitation of article capacity, in this article we only introduce the approach to calculating the similarity between semantic fragments based on factual knowledge description. In the ostis-systems, the sc-agent is the only entity that performs transformations and information processing in sc-memory. Therefore, this article proposes an approach to develop a sc-agent for calculating the similarity between semantic fragments described based on factual knowledge in the ostis-systems [1,2].

I. EXISTING APPROACHES AND PROBLEMS

SPICE (Semantic Propositional Image Caption Evaluation) approach is mainly used to evaluate the quality of automatically generated image caption. The basic principle of this approach is to convert the automatically generated image caption and the image caption labeled by the user into the representation form of the semantic networks, and then calculate the semantic similarity between them. When calculating the semantic similarity, this approach first decomposes the semantic fragments representing the image caption into sub-structures according to the representation rules of knowledge, and then the final similarity is obtained according to the matching relationship of the sub-structures between the semantic fragments.

Although the SPICE approach calculates the similarity between semantic segments based on semantics, there is no uniform rule standard for constructing the semantic networks in this approach, so this approach cannot convert complex image captions into semantic network forms [1].

II. PROPOSED APPROACH

Based on the SPICE approach and the OSTIS Technology used to develop semantic intelligence systems and the corresponding platforms, tools and approaches, an approach to developing a sc-agent for calculating the similarity between semantic fragments described based on factual knowledge is proposed in this article. The main function of this sc-agent is to use the semantic approach to calculate the similarity between any two semantic fragments described based on factual knowledge in the knowledge base of the ostis-systems. Next, we will introduce in detail the development process of the sc-agent for calculating the similarity between semantic fragments described based on factual knowledge [2].

The sc-agent for computing semantic similarity between semantic fragments described based on factual knowledge needs to complete the following tasks:

1. according to the representation rules of factual knowledge, the standard answers and user answers in the form of semantic networks are decomposed into substructures;
2. using formulas (1), (2), and (3) to calculate the precision P_{sc} , recall R_{sc} and similarity F_{sc} between semantic fragments.

$$P_{sc}(u, s) = \frac{|T_{sc}(u) \otimes T_{sc}(s)|}{|T_{sc}(u)|} \quad (1)$$

$$R_{sc}(u, s) = \frac{|T_{sc}(u) \otimes T_{sc}(s)|}{|T_{sc}(s)|} \quad (2)$$

$$F_{sc}(u, s) = \frac{2 \cdot P_{sc}(u, s) \cdot R_{sc}(u, s)}{P_{sc}(u, s) + R_{sc}(u, s)} \quad (3)$$

The main calculation parameters in the formulas include:

- $T_{sc}(u)$ — all substructures after the decomposition of the user answers u ;

- $T_{sc}(s)$ — all substructures after the decomposition of the standard answers s ;
- \otimes — binary matching operator, which represents the number of matching substructures in the set of two substructures.

Since the sc-agent developed in this article will be used for answer verification in the future, we will regard the two semantic fragments used to calculate similarity as the semantic fragment representing the standard answer and the semantic fragment representing the user answer.

Next, we will introduce the working algorithm of this sc-agent in detail:

Algorithm 1 — The working algorithm of sc-agent for computing semantic similarity between semantic fragments described based on factual knowledge

Input: Semantic fragment of standard answers and semantic fragment of user answer. The condition of the sc-agent response is that two semantic fragments that use factual knowledge to represent the answer appear in the sc-memory, and the similarity between them has not been calculated.

Output: The precision, recall and similarity between fragments, and the sc-node used to record the matching status of substructures.

1. checking whether the fragment of standard answer and the fragment of user answer exist at the same time, if so, go to step 2), otherwise, go to step 10);
2. according to the rules of factual knowledge representation (various types of sc-constructions), the semantic fragments of standard answer and user answer are decomposed into substructures;
3. iteratively traverse each substructure of the standard answer and user answer, classify them according to the type of substructure (three element sc-construction, five element sc-construction, etc.), and count the number of all substructures;
4. one type of substructure is randomly selected from the set of recorded standard answer substructure types;
5. according to the standard answer substructure type selected in step 4), a corresponding type of substructure is selected from the set of recorded user answer substructure types;
6. iteratively compare each substructure with the same substructure type between the standard answer and the user answer, and record the number of matched substructures and the matched substructures. The criterion for judging the matching of the same type of substructures is that the sc-nodes at the corresponding positions between the two

substructures have the same main identifier. If the substructure contains sc-links, the contents of the sc-links at the corresponding positions must be also the same;

7. repeat step 4 — step 6 until all types of substructures have been traversed;
8. using formulas (1), (2), (3) calculate precision, recall and similarity, and generate semantic fragments for recording sc-agent running results;
9. removing all temporary sc-elements created while the sc-agent is running;
10. exit the program.

Figure 1 shows an example of calculating the similarity between semantic fragments.

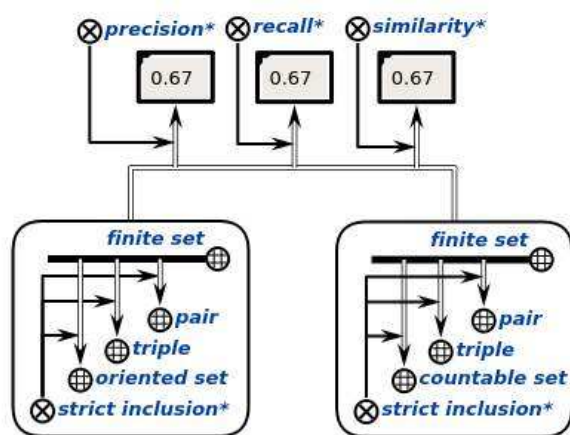


Рис. 1 — An example of calculating the similarity between semantic fragments

III. CONCLUSION

Based on the SPICE approach and OSTIS Technology, the sc-agent for computing semantic similarity between semantic fragments described based on factual knowledge in the ostis-systems is developed in this article. The working algorithm of this sc-agent is introduced in detail in the article. Since a unified knowledge representation approach and knowledge processing tool are provided by OSTIS Technology, the developed sc-agent can calculate the similarity between semantic fragments with very complex semantic structures.

IV. LIST OF REFERENCES

1. Li, W., Qian, L. Development of a problem solver for automatic answer verification in the intelligent tutoring systems / W. Li, L. Qian // International Conference on Open Semantic Technologies for Intelligent Systems. – Minsk, 2021. – P. 169–178.
2. Golenkov, V. V., Guljakina, N. A. “Proekt otkrytoj semanticheskoy tehnologii komponentnogo proektirovanija intellektual’nyh sistem. chast’ 1: Principy sozdaniya project of open semantic technology for component design of intelligent systems. part 1: Creation principles,” / V. V. Golenkov, N. A. Guljakina // Ontologija proektirovanija [Ontology of design]. –2014. –№ 1. – P. 42–64.

IMPLEMENTATION PRINCIPLES OF KNOWLEDGE ACQUISITION FOR INTELLIGENT SYSTEM

Qian Longwei, Li Wenzu
Department of Intelligent Information Technology,
Belarusian State University of Informatics and Radioelectronics
Minsk, Republic of Belarus
E-mail: qianlw1226@gmail.com, wzzggml@gmail.com

The article is dedicated to ontology-based implementation principles in component development of natural language interface, which is able to extract specific construction represented in internal language of intelligent system from unstructured data (i.e., natural language free texts). The main feature of implementation principles is the unified semantic model to integrate different aspects of linguistic knowledge (e.g., syntactic level, semantic level) and various problem solving models oriented on solving relation extraction problems.

INTRODUCTION

The natural language interface is a key component of any intelligent system. It plays the role of information exchange between users and intelligent systems, and realizes the conversion function between external languages and internal languages of intelligent systems. The natural language is one of the most frequently used external language. The natural language text is also the main representation form of unstructured data. For knowledge-based intelligent system, the knowledge base is one of the most key components of intelligent system, it's a finite information structure represented in internal language form. Unstructured data is not organized in any predefined way. It contains a wealth of knowledge of various subject domains. The automatic knowledge acquisition from unstructured data (i.e., natural language text) can effectively improve the efficiency of intelligent system development. The information extraction is the key technology to the implementation of automatic knowledge acquisition.

I. DIRECTION OF INFORMATION EXTRACTION

From the perspective of knowledge acquisition, in this article information extraction is a technology that automatically extracts structured information represented in internal language of intelligent system from unstructured data. For processing of unstructured data, information extraction can be considered as a simplified natural language understanding technology. Information extraction from natural language texts can be divided into the following two directions:

- traditional information extraction focuses on closed domain, i.e., small homogeneous corpora;
- open information extraction, a new domain-independent knowledge discovery paradigm.

In the early stages, there are just limited natural language text about a certain of specific domains. The traditional information extraction methods just extract predefined relation sets including entities, relation and so on. With the

advent of the Internet and Wikipedia, there are massive, multi-source and heterogeneous corpora such as various websites. The target of open information extraction is to extract various relation sets without requiring a pre-specified vocabulary to specify relation type from massive and heterogeneous text corpora, where target relations are unknown in advance. In order that information extraction methods are able to solve this problem, recently many efforts have been invested in exploring methods for open information extraction, which aims to discover new relation types from open-domain corpora.

II. OVERVIEW OF EXISTING APPROACHES

Generally speaking, the procedure of information extraction is divided into entity extraction, relation extraction, attribute extraction and entity linking and so on. In this article, we just consider the open relation extraction. The open relation extraction approaches directly extract relation sets that contains entities and relations from open-domain text corpora.

Several open information extraction systems had developed to solve the problem. For example, TextRunner [1] and other open information extraction system achieve promising performance in open relation extraction on English sentences. They realize relation extraction which verb phrases or non-verb phrases can be considered as, and the binary relation, multiple relation extraction. For Chinese sentences that are very different from English, according to the characteristics of the Chinese language, there are several systems developed to extract relation sets from Chinese sentences, such as CORE [2]. However, The development of these systems lack a unified basic to achieve reasonable compatibility between relation extraction components and intelligent system.

III. PROPOSED APPROACH

As described above, the main difficulties of information extraction come from two aspects:

- the structure analysis of external language texts;

- the analysis of fragment of knowledge base represented in internal language.

In order to solve these two aspects problems, we proposed use OSTIS Technology [4] to develop relation extraction component, which is able to extract sc-construction from natural language texts. The ostis-system is a knowledge-based intelligent system being developed based on OSTIS technology. The sc-construction represented in SC-code, which is an internal language used for knowledge representation, is a fragment of the knowledge bases of ostis-systems. SC-code is the form of semantic network language with the basic set-theoretic interpretation. There are several ways of external representation such as SCg, SCn, SCs, which will be shown blow.

By analyzing the structure of the knowledge base, we can clarify our extraction targets. In order to convert constructions of the external language into the sc-constructions, it is necessary to describe the syntactic and semantic knowledge for text analysis, as well as the construction of extraction rules. Since each external language can have its own set of unique features, within the framework of OSTIS Technology, concept **Subject Domain** is proposed to describe knowledge of a certain of specific domain. In our proposed implementation principles, we attempt to construct ontologies of specific natural language to extract open relations from open text corpora.

Based on the principles of OSTIS Technology, An ontology is interpreted as a specification of the system of concepts of the corresponding subject domain. In each subject domain various distinguished ontologies are described to reflect a certain set of the concept features of the subject domain, for example, terminological ontology, logical ontology, set-theoretic ontology, etc. The general structure of **Subject domain of Chinese language texts** is presented in SCn-language.

Section. Subject domain of Chinese language texts

⇒ *section decomposition**:

- *Section. Subject domain of lexical analysis*
- *Section. Subject domain of syntactic analysis*
- *Section. Subject domain of semantic analysis*

The *Section. Subject domain of lexical analysis*, the *Section. Subject domain of syntactic analysis* and the *Section. Subject domain of semantic analysis* describe specification of a system of concepts, logical rules (e.g., extraction rules) and even other knowledge from the lexical aspect, syntactic aspect and semantic aspect of the Chinese language, respectively.

Aspect that construction of ontologies, the problem solvers of ostis-system also need to be considered, it can utilizes the knowledge base of linguistics to extract sc-construction from natural language free texts based on a series of text processing technologies, i.e., lexical analysis, syntactic analysis, semantic analysis, and extraction rules. The problem solver is a hierarchical system of agents of knowledge processing which oriented on solving specific problem. The general structure for developed problem solver for translating external texts into the fragment of knowledge base is represented below in SCn-language.

Problem solver translating external texts into the fragment of knowledge base

⇐ *decomposition of an abstract sc-agent**:

- ```
{
 • Abstract sc-agent of lexical analysis
 • Abstract sc-agent of syntactic analysis
 • Abstract sc-agent of semantic analysis
 • Abstract sc-agent of generating
 sc-construction
 • Abstract sc-agent of verifying the knowledge
 base
}
```

The structure of developed problem solver corresponds to conversion steps from natural language sentence to sc-construction represented in SC-code.

## IV. CONCLUSION

The article proposes a unified semantic model to solving open relation extraction problem. According to the constructed knowledge base and problem solvers, proposed implementation principles can integrate various types of knowledge and problem solving models. The constructed knowledge base contains various linguistic ontologies that used to analyse natural language texts, not just natural language texts of specific domain.

## V. LIST OF REFERENCES

1. Banko, M. Open Information Extraction from the Web / M. Banko, M. J. Cafarella, S. Soderland, et al. // Proceedings of the 20th International Joint Conference on Artificial Intelligence. –Hyderabad, 2007, –P. 2670–2676.
2. Tseng, Yuen-Hsien Chinese Open Relation Extraction for Knowledge Acquisition / Yuen-Hsien Tseng, Lung-Hao Lee, , Shu-Yen Lin, et al. // Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers. –Gothenburg, 2014, –P. 12–16.
3. Golenkov V.V., Gulyakina N.A.: Project of open semantic technology of component designing of intelligent systems. Part 1 Principles of creation. / V. V. Golenkov, N. A. Gulyakina // Ontology of designing. – 2014. № 1. – c.42–64.

# MODEL AND ALGORITHM FOR ADAPTIVE SEARCH BY LOGICAL EXPRESSIONS

Savenko A. G., Sherstnev A. S.

Institute of Information Technologies of the Belarusian State University of Informatics and Radioelectronics  
Minsk, Republic of Belarus  
E-mail: savenko@bsuir.by

*The article presents the developed graph model and algorithm for adaptive search by logical expressions using the example of a web application for the selection of employees of IT-companies based on their competencies. The analysis of the performance of the search algorithm by logical expressions for borderline cases is carried out. The flexibility of the search algorithm is shown when building queries by minimizing the logical expression of a search query.*

## INTRODUCTION

Information retrieval is an essential part of any automated system, including information systems related to the processing of text data. The important criteria for evaluating the performance of search algorithms are their speed and relevance. This article proposes a general model of a search algorithm that will allow you to efficiently and flexibly, from the point of view of search queries, determine the subset of data of interest to the user.

### I. DATA STORAGE ORGANIZATION MODEL

From the point of view of the information model, the proposed search engine implements a complex search based on logical expressions. The information database is stored in the form of relationships between entities, the combination of which allows you to achieve the desired search criterion. Entities should describe as little information as possible in order to provide less granularity and therefore provide more accurate results. Links should be one-way and directional, and their number should be minimal. The described data storage structure can be easily implemented as a graph database. An illustration of a graph database model using the example of an information system for the selection of employees of an IT-company to work on project tasks [1] in accordance with their competencies is shown in figure 1.

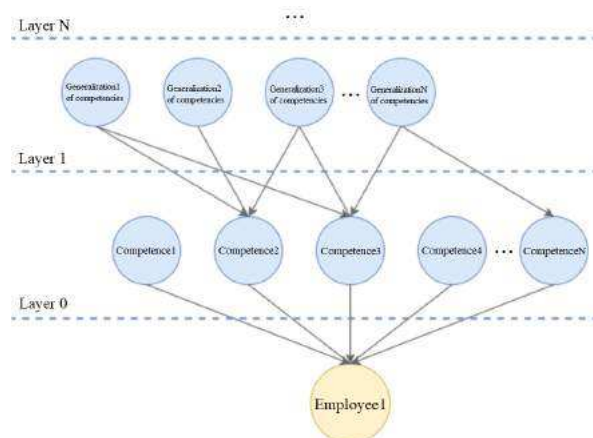


Рис. 1 – Sample graph database model

As you can see from the sample database model, the data is organized in layers. At the lowest level is the employee information node. This node is linked to skill nodes by one-way communication. In turn, skill nodes are associated with the next layer of generalizations of these skills into broader groups of concepts. This generalization will allow the search system to find employees according to broader concepts of their competencies. Also, on one conditional layer, links between nodes are prohibited to prevent cyclic search.

### II. SEARCH ALGORITHM

The adaptive search in the proposed system is carried out by logical expressions. A search query can include basic logical operations such as AND, OR, as well as a grouping operation with priority, and the search process itself is a graph traversal. However, first, it is necessary to transform the input expression into a disjunctive normal form (DNF), where in the given example, the competencies of employees act as logical literals. DNF will allow converting any incoming logical expression into a disjunction of conjunctions of literals, which will make it possible to split the algorithm into two stages:

1. Search for all employees with several competencies at the same time:
  - the first step is to convert the original expression to DNF;
  - the next steps are a cycle through all conjunctive groups and the formation of queries to traverse the graph, taking into account the inclusion of all competencies from the group.
2. Combining all results from the previous step and removing duplicates.

DNF will also allow minimizing a logical expression, which will speed up the search algorithm by reducing the number of logical literals. Two-way breadth first search is used to find paths on a graph.

### III. SEARCH ALGORITHM EFFICIENCY ANALYSIS

An important criterion in the development of an adaptive search algorithm is its performance (since it is necessary to process a large amount of

data in an acceptable user waiting time), as well as flexibility in building search queries. Based on this, we will consider the boundary (worst) cases of the algorithm operation, in which the search will be as ineffective as possible. The number of connections between the nodes of two layers at a given level is determined by the function  $L(k, m, n)$ , where  $k$  is the number of nodes in the first layer,  $m$  is the number of nodes in the second layer,  $n$  is the number of the level, and at the same time  $k > 0; m > 0; n \geq 0; k, m, n \in N$ . The number of nodes in the next layer is set by the function  $G(t, n)$ , where  $t$  is the number of nodes of the previous layer,  $n$  is the number of the level, and at the same time  $t > 0; n \geq 0; t, n \in N$ . The total number of links in a multilayer graph, with restrictions on links, is described by formula 1. The total number of nodes is determined by formula 2. The extreme worst case for a search would be the number of links that needs to be checked from the topmost layer to the bottommost, that is, traverse the entire graph. The function describing this extreme case creates connections between each node of the initial layer and each node of the next one and has the form  $L_{max} = (k, m, n) = km$ . Then the growth rate of the number of links  $R(1, n, L_{max}, G_i)$  and the number of nodes  $E(1, n, G_i)$  at  $n$  levels will have the form shown in Figures 2 and 3.

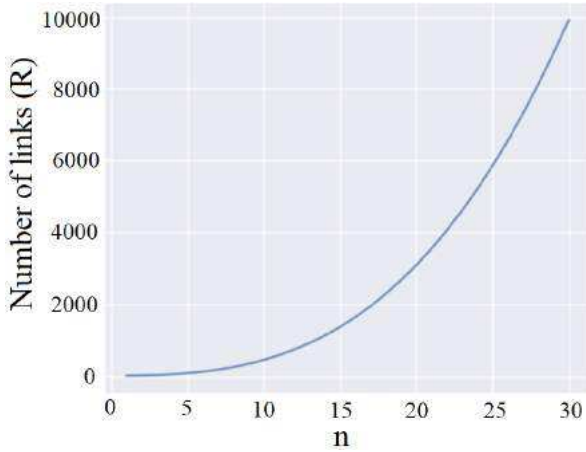


FIG. 2 – Graph of the growth rate of the number of links for  $n$  number of layers

$$R(t, n, L, G) = \begin{cases} (R(G(t, n), n - 1, L, G) + L(t, G(t, n), n), n > 0 \\ 0, n = 0 \end{cases}, \quad (1)$$

where  $t$  is the number of nodes on the initial layer;  $t > 0; t \in N$ ;  $n$  is the total number of layers;  $n \geq 0; n \in N$ ;  $L$  is a function that determines the number of connections between the nodes of two layers at a given level;  $G$  is a function that determines the number of nodes in the next layer.

$$E(t, n, G) = \begin{cases} E(G(t, n), n - 1, G) + G(t, n), n > 0 \\ 0, n = 0 \end{cases}, \quad (2)$$

where  $t$  is the number of nodes on the initial layer;  $t > 0; t \in N$ ;  $n$  is the total number of layers;  $n \geq 0; n \in N$ ;  $G$  is a function that determines the number of nodes in the next layer.

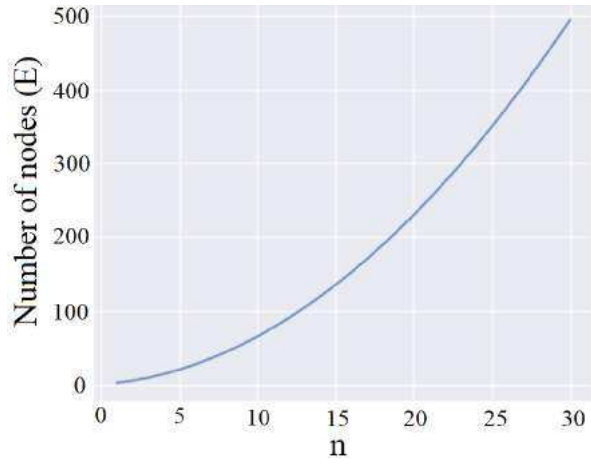


FIG. 3 – Graph of the growth rate of the number of nodes for  $n$  number of layers

Obviously, even with a small number of layers  $n \geq 30$ , the number of nodes and links is quite different in order, and the dependence itself is far from linear and closer to a power-law or exponential.

#### IV. CONCLUSIONS

As a result of the study, a model of information system data organization and a flexible adaptive search algorithm by logical expressions were developed. The data organization model is a graph database. The created algorithm has a high relevance of the results obtained and a high speed of obtaining them. The developed model and search algorithm are applied in the implemented cross-platform system for recruiting employees of IT-companies based on their competencies.

#### V. BIBLIOGRAPHY

1. Savenko, A. G. Web system for the adaptive search for employees of IT-companies based on their competencies by logical expressions / A. G. Savenko, A. S. Sherstnev // Web programming and Internet technologies WebConf2021: materials of the 5th International scientific and practical conference –Minsk, BSU –2021. – P. 127–129.

# SEGMENTATION OF BRAIN TUMOR MULTI-PARAMETRIC MRI SCANS USING ARTIFICIAL NEURAL NETWORKS

Zhirko M., Titko V., Grakova N.  
Department of Intellectual Information Technologies,  
Belarusian State University of Informatics and Radioelectronics  
Minsk, Republic of Belarus  
E-mail: {mzhirko, vtitko27}@gmail.com, grakova@bsuir.by

*In this paper, we present an automated method for brain tumor segmentation. A deep learning-based segmentation algorithm is expected to be able to solve diagnosis making, treatment planning, and resembling tasks. The automated method will help specialists to make more specific analyses in a relatively short amount of time. The analysis, made by this method is considered to have quite a good accuracy. The project is made open-source so that everyone is able to use it and do a contribution to make it better. Convolution Neural Networks (CNNs) are widely used for computer tomography (CT) images. Reliable, fast, and fully automated brain tumor segmentation is based on a Dense V-Network. An ample clinically acquired multi-parametric MRI dataset is used for the training, validation, and testing of our model. We conclude that deep-learning-based segmentation represents an acceptable method for brain tumor abdominal CT segmentation whose accuracy can surpass current methods.*

## INTRODUCTION

Segmenting abdominal images can facilitate clinical workflows in multiple domains, encompassing diagnostic interventions, treatment planning, and treatment delivery. Due to its clinical prevalence, brain tumor CT has been the focus of such research, along with three methods: statistical models (SM), multi-atlas label fusion (MALF), and registration-free methods.

Manual segmentation of 3D abdominal images is labor-intensive and impractical for most clinical workflows, therefore the need for automatic segmentation tools has emerged. In registration-free methods, the challenges consist of constructing variable- and deformation-invariant features ("hand-tuned" or learned) that characterize anatomy from training data sets that are not registered. However, these techniques have been shown to yield less accurate multi-organ segmentation than the registration-based approaches.

## I. RELATED WORK

The main methodologies for brain tumor segmentation include the usage of different algorithms. For example, statistical approaches, that involve point distribution models or conditional shape modeling. Or multi-atlas label fusion methods could be used. But those methods are limited by image registration accuracy. It happens because of the huge variability of brain tumors. They could differ in size, shape, position, and appearance.

To avoid that problem we can use registration-free methods. Some of those could be based on voxel-by-voxel classification. Other approaches could use handcrafted features or feed-forward neural networks. But lots of recent works involve classifiers which are based on fully convolutional neural networks (FCNN).

With fully convolutional neural networks, there are two main approaches. In the first approach we're segmenting volumetric medical images slice-by-slice. Research shows that it works well, but results between adjacent slices of the same MRI or CT aren't always consistent. This problem can be partly solved with the passing of the previous slices mask as additional input to FCNN. But still, it doesn't make the whole use of data structure. 3D FCNN was proposed to solve this issue.

The second approach uses the whole volumetric medical image as an input to 3D FCNN. With help of 3D layers (convolution, pooling, up-sampling, etc.), this model can successfully exploit the volumetric nature of this data. The training algorithm for this network is very similar to the training algorithm of a simple neural network. Also, as experiments show, those type of networks doesn't need a lot of data. In the original U-Net paper, the authors used just 30 images.

## II. DATA

Ample multi-institutional routine clinically acquired multi-parametric MRI (mpMRI) scans of glioma, with pathologically confirmed diagnosis and available MGMT promoter methylation status, are used as the training, validation, and testing data for this method. All Brain Tumor Segmentation (BraTS) mpMRI scans describe native (T1) and post-contrast T1-weighted (T1Gd), T2-weighted (T2), and T2 Fluid Attenuated Inversion Recovery (T2-FLAIR) volumes, and were acquired with different clinical protocols and various scanners from multiple data contributing institutions.

All the imaging datasets have been manually annotated, by one to four raters, following the same annotation protocol, and their annotations were approved by experienced neuro-radiologists. Annotations comprise the GD-enhancing tumor, the peritumoral edematous/invaded tissue, and the necrotic tumor core. The ground truth data were

created after their pre-processing, i.e., co-registered to the same anatomical template, interpolated to the same resolution (1 mm<sup>3</sup>), and skull-stripped.

### III. OUR METHOD

For our task we used the Dense V-network. This method uses a fully convolutional neural network, which is inspired by U-Net architecture. The critical features of this approach are the following:

- A batch-wise spatial dropout;
- Dense feature stacks;
- V-network downsampling and upsampling;
- Dilated convolution;
- Explicit spatial prior.

### IV. TRAINING

The network was trained for 42 epochs using Adam optimizer with an initial learning rate equal to 0.003. Dice function was used as a loss function, because it considers the loss information both locally and globally, which is critical for high accuracy.

In order not to struck in a local minimum of the loss function, we multiplied the learning rate by a factor of 0.5, when validation loss didn't improve for 5 epochs straight.

Dataset was randomly split into train and validation sets (70% and 30% out of total data respectively). Every MRI image size was set to (184, 184, 128). The batch size was set to 2. Loss changing during the training process is presented on plot 1

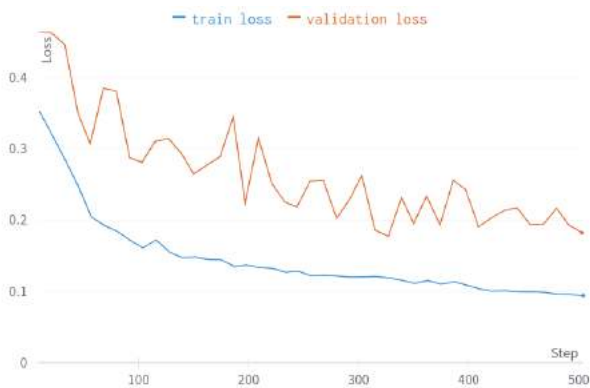


Рис. 1 – Model train and validation loss during training process

### V. RESULTS

Metrics result is provided in table X. Dice score significantly (on 0.16) drops on validation images. It happens because of the variability of different MRI images. Not only do brains and tumors differ in shape and size. Even the MRI images themselves have different resolutions and orientations. In further research, this problem can

be solved by applying more augmentations, and by more sophisticated data preprocessing.

Таблица 1 – Dice score

| Data split | Dice value |
|------------|------------|
| train      | 0.8956     |
| validation | 0.7359     |

Examples of segmentation on two different MRI images are provided below.

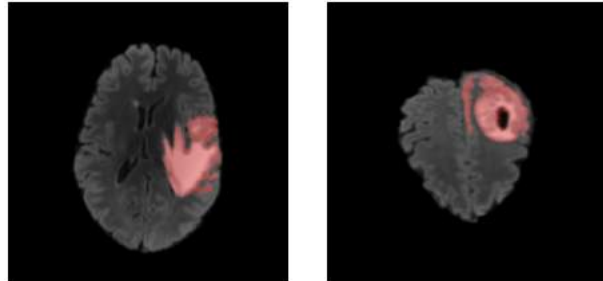


Рис. 2 – Examples of tumor segmentation

### VI. CONCLUSION

Segmentation of brain tumors is proposed to facilitate intraprocedural navigation of brain tumor interventions. The proposed deep learning-based Dense V Net can segment brain tumors more accurately than previous methods. However clinically acceptable segmentation accuracies have yet to be defined for guiding abdominal interventions and depends on the intervention and guidance system. The automatically generated segmentations of abdominal anatomy have the potential to support image-guided navigation in brain tumor endoscopy procedures.

1. Cerrolaza J. J. et al. Automatic multi-resolution shape modeling of multi-organ structures //Medical image analysis. – 2015. – T. 25. – №. 1. – C. 11-21.
2. Okada T. et al. Abdominal multi-organ segmentation from CT images using conditional shape–location and unsupervised intensity priors //Medical image analysis. – 2015. – T. 26. – №. 1. – C. 1-18.
3. Ronneberger O., Fischer P., Brox T. U-net: Convolutional networks for biomedical image segmentation //International Conference on Medical image computing and computer-assisted intervention. – Springer, Cham, 2015. – C. 234-241.
4. Dolz J. et al. Deep CNN ensembles and suggestive annotations for infant brain MRI segmentation //Computerized Medical Imaging and Graphics. – 2020. – T. 79. – C. 101660.
5. Menze B. et al. Multimodal brain tumor image segmentation. benchmark: change detection //Proceedings of MICCAI-BRATS. – 2016.
6. Zhou X. et al. Three-dimensional CT image segmentation by combining 2D fully convolutional network with 3D majority voting //Deep Learning and Data Labeling for Medical Applications. – Springer, Cham, 2016. – C. 111-120.



Секция «Проектирование  
встраиваемых систем»

Section «Embedded systems design»

# УСТРОЙСТВО ДЛЯ МОНИТОРИНГА РАДИОЧАСТОТНОГО СПЕКТРА НА БАЗЕ XILINX ZYNQ

Азаров И.С., Качинский М.В., Петровский Н.А.,  
Рыбенков Е.В., Станкевич А.В.,

Кафедра электронных вычислительных средств,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь

E-mail: {azarov, kachinsky, stankevich, nick.petrovsky, rybenkov}@bsuir.by

*Рассматривается реализация аппаратного HDL-описания и программной части устройства мониторинга радиочастотного спектра на базе ПЛИС Xilinx Zynq Z7045 и трансивера Analog Devices ADRV9375.*

## ВВЕДЕНИЕ

Мониторинг радиочастотного спектра позволяет оценить качество сигналов и утилизацию доступного диапазона частот, обнаружить в сигналах помехи, искажения уровня и изменения частот передачи.

### 1. ПРОЕКТИРОВАНИЕ ПРОТОТИПА

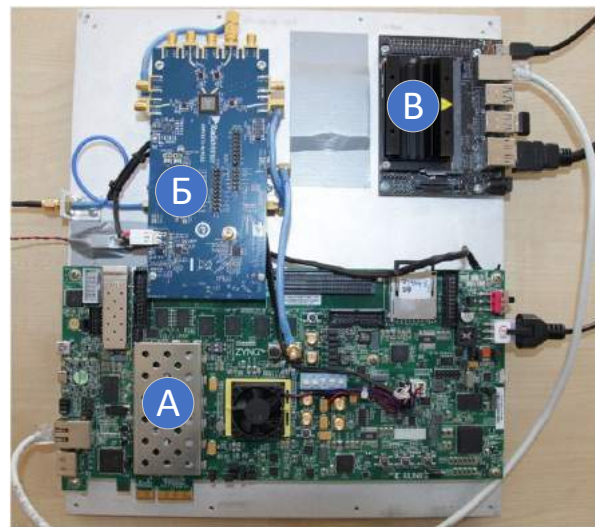
Особенностью устройства является необходимость обработки сигналов широкого частотного диапазона от 300 МГц до 2 ГГц с высоким разрешением по уровню (16 двоичных разрядов). Устройство должно осуществлять сканирование указанного диапазона частот, причем должна иметься возможность установки более узкого диапазона сканирования для более детального изучения спектра.

Для сокращения сроков разработки макет приемного устройства мониторинга радиочастотного спектра был реализован по методологии быстрого проектирования на базе отладочных плат [1]. Основным узлом макета для приема сигнала является радиочастотный трансивер Analog Devices ADRV9375, обладающий следующими характеристиками:

- перестраиваемый диапазон 300–6000 МГц;
- полоса приемника: 8–100 МГц;
- 2 канала приемника;
- цифровой интерфейс JESD204B для взаимодействия с ПЛИС.

В модуле ADRV9375 задействуется приемник, с помощью которого осуществляется сканирование заданного рабочего диапазона частот. Для настройки модуля ADRV9375, первичной обработки принятых данных, вычисления спектра и передачи полученного радиочастотного спектра в систему отображения через интерфейс Ethernet используется отладочная плата ZC706. Плата выполнена на базе системы на кристалле Xilinx XC7Z045 (совмещает в себе двухъядерный процессор архитектуры ARM Cortex A9 и ПЛИС серии Kintex-7). Радиомодуль ADRV9375 подключается к модулю ZC706 с помощью интерфейса FPGA Mezzanine Card (FMC согласно спецификации VITA 57.1). Для анализа и визуализации радиочастотного спектра используется

плата Jetson Nano, представляющая собой одноплатный ARM-компьютер со 128-ядерным графическим ускорителем. Внешний вид макета в сборе приведен на рисунке 1.



A – ZC706; Б – ADRV9375; В – Jetson Nano

Рис. 1 – Макет устройства для мониторинга радиочастотного спектра

Вычисление спектра реализовано аппаратно на базе программируемой логики кристалла Xilinx XC7Z045 установленного на плате ZC706. Для сокращения сроков разработки в качестве базового проекта, использующего аппаратную платформу Xilinx ZC706 + ADRV9375, был выбран пример проектирования (Reference Design) Oscilloscope (двухканальный осциллограф) [2]. Для реализуемой задачи второй канал приемника RX2 ADRV9375 не используется, поэтому можно вместо него подать выходные данные амплитудного (вместо синфазной составляющей  $i$ ) и фазового (вместо квадратурной составляющей  $q$ ) спектров. Такой подход позволит не менять конфигурацию DMA и использовать готовые драйвера взаимодействия с аппаратной частью.

### II. АППАРАТНОЕ ОБЕСПЕЧЕНИЕ

Структурная схема блока цифровой обработки сигнала приемника радиочастотного трансивера ADRV9375 приведена на рисунке 2.

Размер обрабатываемого фрейма составляет 4096 отсчетов. Входной сигнал от АЦП приемника RX1 AD9375 поступает на умножитель для взвешивания окном Хеннинга, симметричная половина коэффициентов которого хранится в ПЗУ. Параллельно обрабатываются действительная и мнимая части входного сигнала, которые затем поступают в модуль БПФ. В качестве модуля БПФ использовалось IP-ядро Xilinx Fast Fourier Transform v9.1, со следующими параметрами: количество точек – 4096; разрядность данных для  $Re(\cdot)$  и  $Im(\cdot)$  по отдельности, входные – 16; выходных – 32 разряда;



Рис. 2 – Структурная схема блока цифровой обработки сигналов

Выходные данные БПФ сохраняются в FIFO-буфере, реализованном на двух модулях двухпортовой памяти с организацией  $4096 \times 32$  раздельно для  $Re(\cdot)$  и  $Im(\cdot)$  частей выходного сигнала.

Модуль CORDIC предназначен для вычисления амплитудного  $|X|$  и фазового  $\phi_x$  спектров. В качестве этого модуля использовалось IP-ядро Xilinx CORDIC v6.0 в режиме преобразования декартовых координат в полярные. Входные данные вещественной и мнимой частей модуля CORDIC – 32-разрядные, выходные – по 16 разрядов для амплитуды и фазы. Выходные значения амплитуды и фазы фрейма накапливаются в выходном FIFO-буфере, выполненном на двух модулях двухпортовой памяти с организацией  $4096 \times 16$ . Наличие FIFO необходимо для согласования скоростей работы модулей БПФ и CORDIC с внешним приемником информации.

Общие аппаратные затраты модифицированного Reference Design макета приемного устройства мониторинга радиочастотного спектра с учетом аппаратного вычисления спектра по отчету системы проектирования Vivado приведены в таблице 1. При системной тактовой частоте 200 МГц время обработки фрейма из 4096 отсчетов составляет около 800 мкс.

Таблица 1 – Аппаратные затраты ресурсов кристалла ПЛИС xc7z045ffg900-2

| Ресурс | Исп.  | %     | Ресурс | Исп.  | %     |
|--------|-------|-------|--------|-------|-------|
| LUT    | 48210 | 22.05 | LUTRAM | 3029  | 4.30  |
| FF     | 57882 | 13.24 | BRAM   | 64.50 | 11.83 |
| DSP    | 56    | 6.22  | IO     | 182   | 50.28 |
| GT     | 4     | 16    | BUFG   | 15    | 46.88 |
| MMCM   | 5     | 62.50 | PLL    | 1     | 12.50 |

### III. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

В результате анализа существующих программных средств для взаимодействия с радиочастотными приёмо-передающими устройствами от Analog Devices, выбран LibIO (Industrial Input Output) [5]. В качестве управляющей операционной системы используется Linux для встраиваемых систем. Для подготовки конфигурационных файлов (ядро ОС, загрузчик, конфигурационные данные FPGA, образ файловой системы) использовались следующие пакеты исходных текстов ПО, с версиями актуальными на момент разработки:

1. ядро ОС Linux (4.14.0) [3];
2. вторичный загрузчик U-Boot 2018.01 [4];
3. плагин трансивера AD9375 для подсистемы ОС Linux, IO [5];
4. дистрибутив BuildRoot [6];
5. RTL описание архитектуры системы FPGA в виде Makefile и скриптов для Tcl/Tk САПР Vivado 2018.2 [2]

Для связи устройства с хост-компьютером на базе Jetson Nano с целью последующего анализа и визуализации радиочастотного спектра разработан сетевой сервис (DCAP – digital capture) для сети Ethernet. Сервис представляет из себя управляющее ПО, которое загружается автоматически с помощью скриптов `init.d` для BusyBox во время запуска отладочной платы XC706. Данный сервис регистрирует сетевой сокет типа TCP-IP для получения управляющих сообщений в формате JSON. После фазы конфигурации сканируемого частотного диапазона, DCAP выполняет передачу амплитудного и фазового спектра в виде UDP-пакетов на заданный порт и адрес. Для хост-компьютера разработано ПО с графическим интерфейсом для управления мониторингом и отображения спектра с использованием Python3 и PyQt.

Испытания разработанного макета подтвердили возможность мониторинга радиочастотного спектра в реальном времени.

#### СПИСОК ИСТОЧНИКОВ

1. Петровский Ал.А., Станкевич А.В., Петровский А.А., Быстрое проектирование систем мультимедиа от прототипа // Петровский Ал.А., Станкевич А.В., Петровский А.А. – Минск: Бестпринт, 2011 – 412 с.
2. HDL Reference Designs: [Электронный ресурс]. – Режим доступа: [https://github.com/analogdevicesinc/hdl/tree/hdl\\_2018\\_r2/library/](https://github.com/analogdevicesinc/hdl/tree/hdl_2018_r2/library/). Дата доступа: 19.11.2021.
3. Linux Kernel: [Электронный ресурс]. – Режим доступа: <https://github.com/analogdevicesinc/linux>. Дата доступа: 19.11.2021.
4. U-boot: [Электронный ресурс]. – Режим доступа: <https://github.com/Xilinx/u-boot-xlnx>. Дата доступа: 19.11.2021.
5. IO library: [Электронный ресурс]. – Режим доступа: <https://github.com/analogdevicesinc/libio>. Дата доступа: 19.11.2021.
6. BuildRoot (embeded linux distro): [Электронный ресурс]. – Режим доступа: <http://git.buildroot.net/buildroot/>. Дата доступа: 19.11.2021.

# СХЕМНАЯ РЕАЛИЗАЦИЯ МОДУЛЯРНЫХ СУММАТОРОВ В ЗАКАЗНЫХ КМОП СБИС

Бибило П. Н., Кириенко Н. А.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: {bibilo, kir}@newman.bas-net.by

Приведены результаты экспериментов по схемной реализации модулярных сумматоров в библиотеке проектирования отечественных заказных КМОП СБИС. Исходные описания проектов модулярных сумматоров описывались системами не полностью определенными (частичными) булевыми функциями. Логическая оптимизация проводилась в классе дизъюнктивных нормальных форм (ДНФ), полиномиальных представлений Рунда-Маллера и BDD представлений булевых функций. Синтезированные схемы оценивались по площади, задержке и энергопотреблению. Установлено, что использование моделей частичных функций и предварительной логической BDD оптимизации позволяет значительно улучшать параметры синтезируемых логических схем.

## ВВЕДЕНИЕ

Создание эффективных параллельных вычислительных систем часто опирается на модулярные вычисления [1] и соответствующие им аппаратные реализации. Схемные реализации модулярных сумматоров для заказных СБИС рассматривались в [2], где были исследованы некоторые способы технологически независимой оптимизации реализуемых сумматорами систем булевых функций. Синтез функциональных комбинационных блоков заказных СБИС в заданном базисе (библиотеке) логических элементов традиционно разбивается на два больших этапа: технологически независимую оптимизацию реализуемых систем булевых функций и технологическое отображение элементов. Решающее влияние на основные параметры (сложность, быстродействие, энергопотребление) логических схем оказывает первый этап. В данной работе предлагается описывать модулярные сумматоры системами частичных булевых функций и осуществлять логическую оптимизацию, выполняя совместную минимизацию функций в классе ДНФ [3, 4], полиномиальных представлений Рунда-Маллера [5] и BDD представлений (Binary Decision Diagrams) [6]. В результате проведенных экспериментов установлено, что использование моделей частичных функций и предварительной логической оптимизации позволяет значительно улучшать параметры синтезируемых логических схем.

## I. ПРИМЕР МОДУЛЯРНОГО СУММАТОРА

Суммирование по модулю  $p_i$  (основание модулярной вычислительной системы) для двух операндов  $a, b$ , находящихся в диапазоне  $\{0, 1, \dots, p_i\}$  выполняется согласно формулам

$$|a + b|_{p_i} = (a + b - p_i), \text{ если } (a + b) \geq p_i,$$

$$|a + b|_{p_i} = (a + b), \text{ если } (a + b) < p_i.$$

В таблице 1 задана система частичных булевых функций  $y_2(a_2, a_1, b_2, b_1)$ ,  $y_1(a_2, a_1, b_2, b_1)$ , задающая сумматор по модулю 3 ( $p_i=3$ ), неопре-

деленные значения булевых функций  $y_2, y_1$  обозначены символом «—».

Таблица 1 – Сумматор по модулю 3

| $a$      | $b$      | $ a + b _3$ |
|----------|----------|-------------|
| $a_2a_1$ | $b_2b_1$ | $y_2y_1$    |
| 0 0      | 0 0      | 0 0         |
| 0 0      | 0 1      | 0 1         |
| 0 0      | 1 0      | 1 0         |
| 0 0      | 1 1      | — —         |
| 0 1      | 0 0      | 0 1         |
| 0 1      | 0 1      | 1 0         |
| 0 1      | 1 0      | 0 0         |
| 0 1      | 1 1      | — —         |
| 1 0      | 0 0      | 1 0         |
| 1 0      | 0 1      | 0 0         |
| 1 0      | 1 0      | 0 1         |
| 1 0      | 1 1      | — —         |
| 1 1      | 0 0      | — —         |
| 1 1      | 0 1      | — —         |
| 1 1      | 1 0      | — —         |
| 1 1      | 1 1      | — —         |

## II. ОПИСАНИЕ ЭКСПЕРИМЕНТОВ

Исходными описаниями логики являлись таблицы истинности систем булевых функций на языке SF в формате SDF (матричных описаний) [6]. В системе FLC-2 [7] выполнялась логическая оптимизация и осуществлялся перевод оптимизированных SF описаний в описания на языке VHDL, в системе CMOSLD [8] выполнялась схемная реализация полученных VHDL описаний в библиотеке проектирования отечественных КМОП СБИС с помощью синтезатора LeonardoSpectrum. Для каждой полученной схемы подсчитывалась площадь схемы (суммарное число транзисторов во всех элементах схемы) и энергопотребление (среднее значение потребляемого тока). Для оценки энергопотребления генерировались псевдослучайные тесты и проводилось схемотехническое моделирование SPICE описаний схем. В каждом из шести эксперимен-

тов синтезировались схемы 14 модулярных сумматоров.

Эксперимент 1. Схемная реализация исходных (неоптимизированных) VHDL описаний систем совершенных ДНФ (СДНФ) полностью определенных функций.

Эксперимент 2. Предварительная логическая оптимизация СДНФ полностью определенных функций в базисе полиномов Рида-Маллера и последующий синтез схемы.

Эксперимент 3. Предварительная логическая оптимизация на основе совместной минимизации в классе ДНФ систем частичных функций [4].

Эксперимент 4. Предварительная логическая оптимизация на основе совместной минимизации систем частичных функций программой Espresso ИС [3].

Эксперимент 5. Предварительная логическая оптимизация в классе BDD представлений частичных функций [8].

Эксперимент 6. Предварительная логическая оптимизация систем СДНФ полностью определенных функций в классе BDDI представлений (с нахождением пар взаимно инверсных подфункций).

### III. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Результаты эксперимента приведены в таблице 2. Имя схемы  $Modp_i$  соответствует сумматору по модулю  $p_i$ . В таблице 2 используются следующие обозначения:  $k$  – число общих элементарных конъюнкций, входящих в СДНФ всех  $m$  функций системы. Для сумматора по модулю 3 (таблица 1)  $k=9$ .

Площади заказных СБИС  $S_1 - S_6$  (для соответствующих экспериментов) подсчитывались в числе транзисторов, средние потребляемые токи  $I_1 - I_6$  в микроамперах (мкА). Символом \* отмечены лучшие решения.

### ЗАКЛЮЧЕНИЕ

Использование моделей частичных функций и предварительной логической оптимизации на основе BDD представлений (эксперимент 5) позволяет получать модулярные сумматоры, характеризующиеся меньшими значениями площади и потребляемого тока.

1. Модулярные параллельные вычислительные структуры нейропроцессорных систем / Н. И. Червяков [и др.]. – М.: Физматлит, 2003. – 288 с.
2. Сравнительное исследование и анализ методов аппаратной реализации сумматоров по модулю [Электронный ресурс] / Е. С. Балака [и др.] // Universum: Технические науки: электрон. научн. журн. 2016. № 1 (23). – Режим доступа: <http://7universum.com/ru/tech/archive/item/2887> – Дата доступа: 25.09.2021.
3. Brayton, K. R. Logic minimization algorithm for VLSI synthesis / K. R. Brayton, G. D. Hachtel, C. T. McMullen, A. L. Sangiovanni-Vincentelli // Boston, e.a.: Kluwer Academic Publishers. – 1984. – 193 p.
4. Торопов, Н. Р. Минимизация систем булевых функций в классе ДНФ / Н. Р. Торопов // Логическое проектирование. – Мн.: Ин-т техн. кибернетики НАН Беларуси. – 1999. – Вып. 4. – С. 4 – 19.
5. Закревский, А. Д. Полиномиальная реализация частичных булевых функций и систем / А. Д. Закревский, Н. Р. Торопов // Минск : Ин-т технической кибернетики НАН Беларуси. – 2001. – 200 с.
6. Бибило, П. Н. Применение диаграмм двоичного выбора при синтезе логических схем / П. Н. Бибило // Минск : Белорусская наука. – 2014. – 231 с.
7. Бибило, П. Н. Система логической оптимизации функционально-структурных описаний цифровых устройств на основе продукционно-фреймовой модели представления знаний / П. Н. Бибило, В. И. Романов // Проблемы разработки перспективных микро- и нанoeлектронных систем. – 2020. – Сб. трудов / под общ. ред. акад. РАН А. Л. Стемковского. – М.: ИППМ РАН. – 2020. – N 4. – С. 9.
8. Система логического проектирования функциональных блоков заказных КМОП СБИС с пониженным энергопотреблением / П. Н. Бибило, В. И. Романов // Микроэлектроника. – 2017. – Т. 46. – № 1. – С.72–88.

Таблица 2 – Результаты экспериментального исследования

| Схема | $k$  | Эксперимент |       |       |       |       |       |       |       |       |       |       |       |
|-------|------|-------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
|       |      | 1           | 2     | 3     | 4     | 5     | 6     | 1     | 2     | 3     | 4     | 5     | 6     |
| mod5  | 25   | $S_1$       | $S_2$ | $S_3$ | $S_4$ | $S_5$ | $S_6$ | $I_1$ | $I_2$ | $I_3$ | $I_4$ | $I_5$ | $I_6$ |
| mod5  | 25   | 200         | 362   | 164   | 202   | *128  | 218   | 14    | 24    | 17    | 14    | *12   | 19    |
| mod7  | 49   | 352         | 482   | 3064  | *228  | 246   | 260   | 27    | 43    | 31    | *23   | 24    | 27    |
| mod9  | 81   | 554         | 1084  | 598   | 598   | *390  | 524   | *28   | 55    | 51    | 39    | 34    | 33    |
| mod15 | 225  | 1154        | 1392  | 1136  | 872   | *310  | 7128  | 66    | 96    | 94    | 68    | *37   | 71    |
| mod17 | 289  | 1502        | 2660  | 1642  | 1802  | *768  | 938   | *51   | 111   | 125   | 92    | 74    | 65    |
| mod19 | 361  | 1714        | 3742  | 1800  | 2106  | *458  | 830   | 59    | 141   | 137   | 103   | *42   | 58    |
| mod23 | 529  | 3530        | 5624  | 2354  | 2368  | *480  | 980   | 117   | 295   | 167   | 138   | *51   | 79    |
| mod25 | 625  | 4478        | 7640  | 3376  | 3430  | *722  | 1134  | 134   | 345   | 222   | 196   | *71   | 105   |
| mod27 | 729  | 6082        | 6352  | 3578  | 3656  | *642  | 1342  | 167   | 359   | 233   | 217   | *67   | 113   |
| mod29 | 841  | 6190        | 6282  | 3826  | 3448  | *702  | 1328  | 168   | 332   | 252   | 205   | *74   | 126   |
| mod31 | 961  | 8634        | 3582  | 3460  | 2820  | *500  | 1234  | 225   | 171   | 215   | 189   | *56   | 118   |
| mod37 | 1369 | 13356       | 12354 | 6444  | 6456  | *882  | 1948  | 369   | 547   | 418   | 321   | *79   | 138   |
| mod59 | 3481 | 36608       | 16012 | 13672 | 12040 | *892  | 2652  | 817   | 512   | 733   | 700   | *100  | 221   |
| mod61 | 3721 | 38560       | 15144 | 13060 | 11654 | *996  | 2246  | 813   | 541   | 705   | 683   | *107  | 218   |

# НЕРАЗРУШАЮЩЕЕ ТЕСТИРОВАНИЕ ЗАПОМИНАЮЩИХ УСТРОЙСТВ НА БАЗЕ ДВОЙНЫХ АДРЕСНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ

Деменковец Д. В., Леванцевич В. А.

Кафедра программного обеспечения информационных технологий,  
Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: Demenkovets@bsuir.by, Lvn@bsuir.by

*Анализируется эффективность применения классических неразрушающих тестов для тестирования запоминающих устройств (ЗУ) и их основные недостатки, среди которых выделяют большую временную сложность и низкую диагностическую способность. Определяется понятие двойной адресной последовательности 2A и приводится пример ее формирования на базе счетчиковых адресных последовательностей. Синтезируется базовый элемент неразрушающих тестов с применением двойных адресных последовательностей. Приводятся два неразрушающих теста ЗУ March\_2A\_1 и March\_2A\_2 для которых оценивается их временная сложность и эффективность обнаружения неисправностей ЗУ. Показывается существенно меньшая временная сложность предложенных тестов и высокая диагностическая способность по сравнению с классическими неразрушающими тестами.*

## ВВЕДЕНИЕ

Важным достижением в развитии функционального диагностирования запоминающих устройств является разработка и применение методов неразрушающего тестирования [1–3]. Основное свойство данных методов заключается в сохранении данных хранимых в памяти после проведения процедуры тестирования. Одной из первых технологий, основанной на применении сигнатурного анализа, была предложенная Б. Конеманом, которая имеет ряд существенных недостатков [1]. Дальнейшим развитием неразрушающего тестирования ЗУ стала методика, предложенная М. Николаидисом, которая основывается на применении классических маршевых тестов [3, 4]. Однако, реализация подобных неразрушающих тестов требует существенного увеличения их временной сложности. Также, данная методика из-за эффекта маскирования не позволяет получить 100%-ую покрывающую способность даже для однократных неисправностей [1, 5]. В тезисе описывается подход, для построения неразрушающих тестов, основанный на применении адресных последовательностей с двукратным повторением адресов [6].

### I. АНАЛИЗ ЭФФЕКТИВНОСТИ ТЕСТОВ ЗУ

Следует выделить две особенности неразрушающих тестов. Во-первых, неразрушающие тесты строятся на базе классических маршевых тестов [1, 2]. Во-вторых, все существующие тесты рассматриваются для ЗУ, содержащих N однократных запоминающих элементов (ЗЭ). В общем случае маршевый тест включает в себя коллекцию маршевых элементов, которые называются фазами теста [4].

Для формального описания неисправных состояний ЗУ используются математические мо-

дели их неисправностей, отражающие реальные физические дефекты ЗУ [2-4, 7].

Наиболее сложными моделями неисправностей, обнаруживаемых маршевыми тестами, являются кодочувствительные неисправности – PSFk (pattern sensitive faults) и их модификации в виде граничных кодочувствительных неисправностей NPSFk. Среди неисправностей NPSFk выделяют пассивные NPSFk (PNPSFk) в которых содержимое базовой ячейки нельзя изменить в зависимости от определенного набора в любых k из N ячеек [3, 4].

В (1) пример неразрушающего теста March Y по методике Николаидиса:

$$\{\uparrow (rb, r\bar{b}); \downarrow (r\bar{b}, rb); \uparrow\downarrow (rb)\}$$

$$\{\uparrow (rb, w\bar{b}, r\bar{b}); \downarrow (r\bar{b}, wb, rb); \uparrow\downarrow (rb)\} \quad (1)$$

В неразрушающем тесте, представленном в (1), b принимает произвольное значение 0 или 1, а  $\bar{b}$  инверсное по отношению к b значение. Сам тест состоит из двух частей, начального теста для получения эталонной сигнатуры  $S_F$  и базового теста, для получения реальной сигнатуры  $S_R$ . После этого две сигнатуры сравниваются. Несоответствие сигнатур свидетельствует о неисправном состоянии памяти. Данный маршевый тест March Y обнаруживает только 4 из 8 видов неисправностей взаимного влияния, это составляет 50 процентов полноты покрытия неисправностей  $CF_{id}$ .

### II. НЕРАЗРУШАЮЩИЕ МАРШЕВЫЕ ТЕСТЫ НА БАЗЕ ДВОЙНЫХ АДРЕСНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ

Для повышения эффективности неразрушающего тестирования ЗУ и построения неразрушающих тестов предлагается использование

модифицированных адресных последовательностей. А именно использование двойных адресных последовательностей. В качестве примера в таблице 1 приведены адресные последовательности для случая исходной счетчиковой последовательности  $A_C$  для  $m = 4$  и ее модификация (A2). Эти модификации называют двойными адресными последовательностями (2A), в силу того что каждый адрес формируется дважды.

Таблица 1 – Адресные последовательности  $A_C$ , и 2A $C$  модификации

| $A_C =$<br>$c_3c_2c_1c_0$ | $2A_C =$<br>$c_3c_2c_1$ | $2A_C =$<br>$c_3c_2c_0$ | $2A_C =$<br>$c_3c_1c_0$ | $2A_C =$<br>$c_2c_1c_0$ |
|---------------------------|-------------------------|-------------------------|-------------------------|-------------------------|
| 0000 (0)                  | 000 (0)                 | 000 (0)                 | 000 (0)                 | 000 (0)                 |
| 0001 (1)                  | 000 (0)                 | 001 (1)                 | 001 (1)                 | 001 (1)                 |
| 0010 (2)                  | 001 (1)                 | 000 (0)                 | 010 (2)                 | 010 (2)                 |
| 0011 (3)                  | 001 (1)                 | 001 (1)                 | 011 (3)                 | 011 (3)                 |
| 0100 (4)                  | 010 (2)                 | 010 (2)                 | 000 (0)                 | 100 (4)                 |
| 0101 (5)                  | 010 (2)                 | 011 (3)                 | 001 (1)                 | 101 (5)                 |
| 0110 (6)                  | 011 (3)                 | 010 (2)                 | 010 (2)                 | 110 (6)                 |
| 0111 (7)                  | 011 (3)                 | 011 (3)                 | 011 (3)                 | 111 (7)                 |
| 1000 (8)                  | 100 (4)                 | 100 (4)                 | 100 (4)                 | 000 (0)                 |
| 1001 (9)                  | 100 (4)                 | 101 (5)                 | 101 (5)                 | 001 (1)                 |
| 1010(10)                  | 101 (5)                 | 100 (4)                 | 110 (6)                 | 010 (2)                 |
| 1011(11)                  | 101 (5)                 | 101 (5)                 | 111 (7)                 | 011 (3)                 |
| 1100(12)                  | 110 (6)                 | 110 (6)                 | 100 (4)                 | 100 (4)                 |
| 1101(13)                  | 110 (6)                 | 111 (7)                 | 101 (5)                 | 101 (5)                 |
| 1110(14)                  | 111 (7)                 | 110 (6)                 | 110 (6)                 | 110 (6)                 |
| 1111(15)                  | 111 (7)                 | 111 (7)                 | 111 (7)                 | 111 (7)                 |

Основная идея неразрушающих маршевых тестов на базе двойных адресных последовательностей основана на том, что при двукратном инвертировании содержимого ячейки ЗУ его значение останется прежним. В соответствии с этим свойством строится базовый элемент:

$$2 \uparrow (rb, w\bar{b}, rb) \quad (2)$$

Отмечается, что использование в базовом элементе двойной адресной последовательности приводит к тому, что каждая ячейка ЗУ последовательно выполнит два перехода из  $b$  в  $\bar{b}$  и, наоборот, из  $\bar{b}$  в  $b$ , сохранив, таким образом, свое начальное значение. Базовый элемент (2) на основе двойных адресных последовательностей позволяет синтезировать два неразрушающих маршевых теста March\_2A\_1 (3) и March\_2A\_2 (4):

$$\{\uparrow\downarrow (rb); 2 \uparrow (rb, w\bar{b}, rb); \uparrow\downarrow (rb)\}. \quad (3)$$

$$\{\uparrow\downarrow (rb); 2 \uparrow (rb, w\bar{b}, rb); 2 \downarrow (rb, w\bar{b}, rb); \uparrow\downarrow (rb)\}. \quad (4)$$

Первая фаза этих тестов получения эталонной сигнатуры  $S_F$ , а их последняя фаза для получения реального значения сигнатуры  $S_R$  после выполнения предыдущих базовых элементов. В случае проявления неисправностей в ходе выполнения базовых элементов их наличие будет определяться выполнением неравенства реальной  $S_R$  и эталонной сигнатуры  $S_F$ .

### III. АНАЛИЗ ЭФФЕКТИВНОСТИ НЕРАЗРУШАЮЩИХ ТЕСТОВ MARCH\_2A\_1 И MARCH\_2A\_2

При допущении, что при реализации теста March\_2A\_1 исходное содержимое ЗУ нулевое, можно заключить об эквивалентности тестов March\_2A\_1 и March Y. Они эквивалентны, как во временной сложности равной  $8N$ , так и в покрывающей способности различных типов неисправностей. Для простейших неисправностей тест March\_2A\_1, в отличие от известных неразрушающих тестов, имеет максимальную диагностическую способность. Также максимально возможная диагностическая способность теста March\_2A\_1 достигается и для случая сложных кодочувствительных неисправностей PNPSFk [6]. Неразрушающий маршевый тест March\_2A\_2 отличается от March\_2A\_1 наличием второго базового элемента с обратным порядком адресов 2A, что расширяет его возможности обнаруживать сложные неисправности, сохраняя эффективность March\_2A\_1 для более простых неисправностей [6].

### IV. ЗАКЛЮЧЕНИЕ

Приведенный анализ свидетельствует о высокой покрывающей способности предложенных тестов, которая сравнима с покрывающей способностью неразрушающего теста March Y для March\_2A\_1, и эффективностью двукратного теста March Y по отношению к March\_2A\_2. Кроме того, наличие базового элемента (2) в обоих тестах обеспечивает максимальную диагностическую способность, достижимую в рамках маршевых тестов, для простых неисправностей и сложных кодочувствительных PNPSFk.

### V. СПИСОК ЛИТЕРАТУРЫ

1. Nicolaidis M. Theory of transparent BIST for RAMs / M. Nicolaidis // IEEE Transactions on Computers – 1996. – Vol. 45, № 10. – P. 1141–1156.
2. Ярмолик, В. Н. Неразрушающее тестирование запоминающих устройств / В. Н. Ярмолик, И. А. Мурашко, А. Куммерт, А. А. Ивањук // Минск: Бестпринт. – 2005.
3. Ярмолик, В. Н. Маршевые тесты для самотестирования ОЗУ / В. Н. Ярмолик, А. П. Занкович, А. А. Ивањук // Минск: Бестпринт. – 2009.
4. Goor A. J. Testing Semiconductor Memories: Theory and Practice / A. J. Goor // Chichester, UK: John Wiley & Sons; 1991.
5. Yarmolik V. N. Aliasing-Free Signature Analysis for RAM BIST / V. N. Yarmolik, M. Nicolaidis, O. Kebichi // IEEE International Test Conference – 1994 – P. 368-377.
6. Ярмолик В. Н. Неразрушающее тестирование запоминающих устройств на базе двойных адресных последовательностей / В. Н. Ярмолик, И. Мрозек, В. А. Леванцевич, Д. В. Деменковец // Доклады БГУИР. – 2021. – № 19(4). – С. 43–51.
7. Mrozek I., Yarmolik V.N. Multiple Control Random Testing. Fundamenta Informaticae. – 2019; –144(1): – P. 23-43.

# МЕТОДИКА СРАВНЕНИЯ ГЕНЕРАТОРОВ СЛУЧАЙНЫХ ЧИСЛОВЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ ПО РЕЗУЛЬТАТАМ ТЕСТОВ NIST

Заливако С. С., Иванюк А. А.

ООО “СК хайникс мемори солишнс Восточная Европа”

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: sergey.zalivako@sk.com, ivaniuk@bsuir.by

*В работе представлена методика сравнения генераторов случайных числовых последовательностей на основе результатов статистических тестов NIST, которые преобразованы в числовые вектора. Предлагается вычислять метрику косинусного расстояния между полученными векторами, что позволит количественно сравнивать качество разрабатываемого генератора с устройствами, соответствующими стандартам NIST.*

## ВВЕДЕНИЕ

Генераторы случайных числовых последовательностей (ГСЧП) применяются в различных сферах информационных технологий: криптография, статистическая выборка, имитационное моделирование, компьютерные игры и др. В зависимости от области использования физической реализации ГСЧП могут предъявляться различные требования. В частности, для использования в криптографических приложениях ГСЧП должен соответствовать стандартам NIST (National Institute of Standards and Technologies) [1]. В соответствии с тестами NIST, разрабатываемый генератор может либо соответствовать стандарту полностью, либо не соответствовать по определенным качественным критериям (например, существенное различие по количеству нулей и единиц в генерируемой последовательности, коррелированность элементов последовательности и др.).

С другой стороны, в процессе разработки, как правило, генератор изначально не соответствует стандарту NIST, поэтому по мере улучшения характеристик было бы желательно иметь количественную оценку несоответствия. В связи с данной необходимостью был предложен метод сравнения результатов тестирования ГСЧП, основанный на вычислении метрики расстояния между числовыми характеристиками результата статистических тестов NIST. Представленный метод позволяет определить количественные различия между реализациями ГСЧП, качество которых априори различно и определяется их структурой.

## 1. СТАНДАРТ NIST ДЛЯ ГСЧП

В соответствии со стандартами NIST, ГСЧП включает в себя два обязательных элемента: источник случайности (Entropy Source) и генератор псевдослучайных числовых последовательностей (ГПСЧП) (Deterministic Random Bit Generator, DRBG). В свою очередь, источник случайности может быть дополнен блоком вы-

равнивания (Conditioning Component), который, как правило, улучшает характеристики равномерности вырабатываемой случайной последовательности. ГПСЧП может быть реализован как блочный шифр (например, AES), либо хеш-функция (HMAC, SHA-256).

Набор статистических тестов NIST состоит из 15 типов тестов: частотный побитовый тест (Frequency Test), частотный блочный тест (Block Frequency Test), тест на последовательность одинаковых бит (Runs Test), тест на самую длинную последовательность единиц в блоке (Longest Run Test), тест рангов бинарных матриц (Rank Test), спектральный тест (FFT Test), тест на совпадение неперекрывающихся шаблонов (Non-Overlapping Template Test), тест на совпадение перекрывающихся шаблонов (Overlapping Template Test), универсальный статистический тест Маурпера (Universal Test), тест на линейную сложность (Linear Complexity Test), тест на периодичность (Serial Test), тест приблизительной энтропии (Approximate Entropy Test), тест кумулятивных сумм (Cumulative Sums Test), тест на произвольные отклонения (Random Excursions Test), другой тест на произвольные отклонения (Random Excursions Variant Test).

Каждый тест проводится на заданном количестве выборок  $m$  и характеризуется 14 параметрами: количество выборок, на которых р-значение попадает в диапазон  $[0,0; 0,1)$ ;  $[0,1; 0,2)$ ;  $[0,2; 0,3)$ ;  $[0,3; 0,4)$ ;  $[0,4; 0,5)$ ;  $[0,5; 0,6)$ ;  $[0,6; 0,7)$ ;  $[0,7; 0,8)$ ;  $[0,8; 0,9)$ ;  $[0,9; 1,0]$ ; р-значение по результатам статистического теста равномерности р-значений, полученных на  $m$  выборках; бинарное значение результата теста равномерности (0 – не пройден, 1 – пройден); доля выборок  $k$  из  $m$ , прошедших тест; бинарное значение, обозначающее превышение порога по выборкам, прошедшим тест (0 – порог не превышен, т.е. тест был пройден на недостаточном числе выборок, 1 – порог превышен). Таким образом, каждый статистический тест может быть представлен набором из 14 параметров. Каждый тип статистиче-



ского теста может проводиться различное число раз (например, частотный побитовый тест – 1 раз, тест на периодичность – 2 раза, тест на совпадение неперекрывающихся шаблонов – 148 раз). В связи с этим общее число тестов, производимых на сгенерированной последовательности равно 188, следовательно, один запуск генератора характеризуется  $14 \times 188 = 2632$  параметрами.

### II. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ

Качество итоговой случайной последовательности определяется исходным качеством источника случайности, которое усиливается применением ГПСЧП. Соответственно, изменение количественных характеристик источника случайности (например, числа выходных значений, тактовой системной частоты и др.) влияет на результат статистических тестов NIST.

Для проведения экспериментального исследования был разработан ГСЧП, состоящий из 32 одинаковых независимо работающих источни-

ков случайности, реализованных на основе физически неклонированных функций (ФНФ). Выходы источников были поданы на входы 32-выходного ГПСЧП, в результате чего генератор вырабатывал 32-разрядные случайные числа. Были рассмотрены 32 конфигурации источников случайности: один из источников активен, 31 – вырабатывает константный ноль; два активны, 30 – вырабатывают константу; ..., 32 источника активны, ни один не вырабатывает константу. Для каждой конфигурации были получены случайные последовательности, состоящие из 40 млн бит, которые, в свою очередь были разделены на  $m = 100$  выборок для тестирования пакетом NIST. В результате статистического тестирования для каждого из генераторов был получен вектор данных, состоящий из 2632 компонент. Между всеми полученными векторами попарно были вычислены косинусные расстояний, показанные на Рисунке 1.

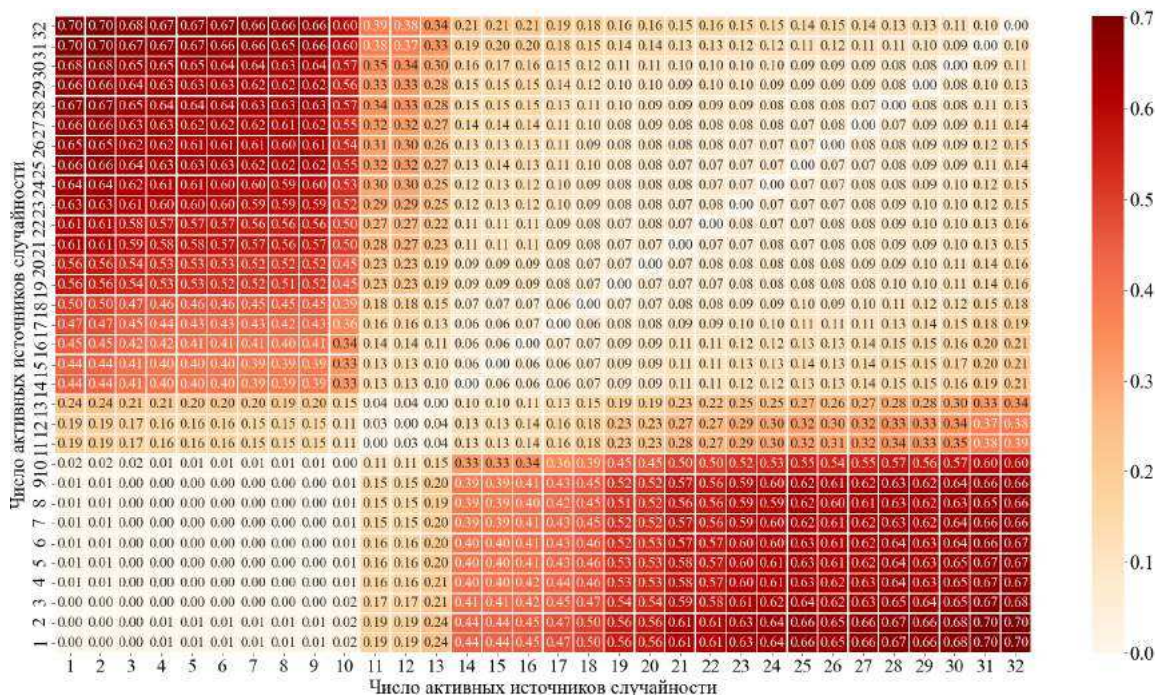


Рис. 1 – Таблица попарных расстояний между результатами тестов NIST

В результате эксперимента было показано, что генераторы с разным числом активных источников случайности существенно отличаются друг от друга. Сравнение результатов NIST-тестов с помощью метрики косинусного расстояния позволяет выделить три группы качества генераторов: низкое качество (генераторы с числом активных источников случайности от 1 до 10), среднее качество (11–13), высокое качество (14–32).

### III. ЗАКЛЮЧЕНИЕ

Предлагаемый метод сравнения ГСЧП на основе вычисления метрики косинусного рассто-

яния между числовыми векторами, характеризующими результаты тестирования NIST, позволяет разработчику количественно оценить разницу текущей версии генератора с устройствами, соответствующими стандартам NIST.

### IV. СПИСОК ЛИТЕРАТУРЫ

1. A statistical test suite for random and pseudorandom number generators for cryptographic applications [Electronic resource]. – Mode of access: <http://csrc.nist.gov/groups/ST/toolkit/rng/documents/SP800-22rev1a.pdf>. – NIST, 2010. – Date of access: 28.10.2021.

# РЕАЛИЗАЦИЯ ФИЗИЧЕСКИ НЕКЛОНИРУЕМОЙ ФУНКЦИИ НА LUT-БЛОКАХ FPGA

Иванюк А. А., Заливако С. С.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники  
ООО “СК хайникс мемори солишнс Восточная Европа”

Минск, Республика Беларусь

E-mail: ivaniuk@bsuir.by, sergey.zalivako@sk.com

*В данной работе рассматривается схемотехническая реализация физически неклоняруемой функции (ФНФ) на LUT-блоках кристаллов FPGA. Новый тип ФНФ может быть использован для построения схем уникальной идентификации цифровых устройств, реализованных на FPGA, и для схем источника цифрового шума для задач генерирования случайных чисел. Предложенная схема ФНФ отличается от существующих решений малыми аппаратными затратами, легко масштабируется под произвольную разрядность и может быть реализована на различных типах FPGA.*

## ВВЕДЕНИЕ

Физически неклоняруемые функции (ФНФ) являются базовыми примитивами для физической криптографии, которая основана на использовании случайных, уникальных, неповторимых, непредсказуемых физических характеристик интегральных схем. Основными областями применения ФНФ является уникальная неклоняруемая идентификация и генерирование истинно случайных числовых последовательностей. На основе этих двух направлений ведутся работы по реализации безопасной аутентификации цифровых устройств, реализации аппаратных водяных знаков и отпечатков пальцев, защите от нелегального использования и клонирования цифровых интегральных схем [1,2].

В данной работе предлагается новая компактная схема ФНФ, сочетающая в себе поведение двух известных типов ФНФ: ФНФ статического ОЗУ и ФНФ кольцевого генератора. Схемотехнически предложенная ФНФ состоит из одного мультиплексора с конфигурацией  $2 \times 1$  и одного инвертора. Показано, что для реализации подобной схемы на FPGA необходим один LUT-блок.

В следующих разделах описана схемотехника предложенной ФНФ, режимы ее функционирования, особенности реализации на FPGA. Приведены экспериментальные данные по функционированию реализованной схемы.

## 1. СХЕМОТЕХНИКА ПРЕДЛАГАЕМОЙ ФНФ

За основу новой схемы ФНФ была взята схема управляемого кольцевого генератора, вырабатывающая выходной сигнал в форме меандра с уникальной невоспроизводимой частотой [1]. В общем случае подобная схема включает в себя схему кольцевого генератора, состоящую из  $2k + 1$  ( $k \in N_0$ ) последовательно соединенных инверторов и цепи обратной связи, включающую управляющий клапан, как правило двухходовой логический элемент И, ко входам которого подключена линия обратной связи генератора и

линия управляющего сигнала  $EN$ . Выход  $Q$  такой схемы может быть подключен к выходному полюсу одного из инверторов в цепи обратной связи. При значении управляющего сигнала  $EN = 1$  схема генерирует импульсы на выходе  $Q$  с уникальной частотой  $F_Q$ , результат анализа которой может быть использован для генерации случайных чисел либо для построения уникального идентификатора. В случае  $EN = 0$  схема прекращает генерацию импульсов и на ее выходе будет сформирован сигнал  $Q = 0$ .

Заменяем управляющий клапан в данной схеме на схему двухходового мультиплексора, как это показано на рисунке 1.

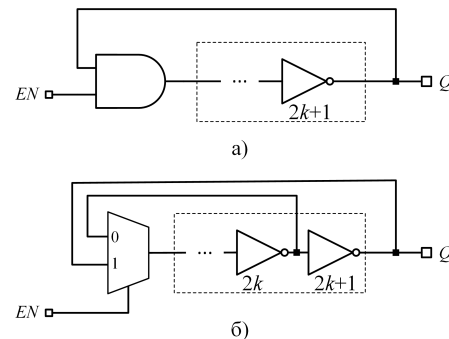


Рис. 1 – Схема управляемого кольцевого генератора (а) и предлагаемая схема ФНФ (б).

Управляющий сигнал  $EN$  подается на селективный вход мультиплексора, который обеспечивает две конфигурации цепи обратной связи: конфигурацию кольцевого генератора ( $EN = 1$ ) и конфигурацию бистабильного элемента ( $EN = 0$ ).

Рассмотрим подробнее функционирование предложенной схемы.

1. *Режим инициализации.* В данном режиме, при удержании  $EN = 0$ , схема в конфигурации бистабильного элемента хранит непредсказуемое значение, которое можно наблюдать на выходе  $Q$ . Поведение схемы в этом режиме аналогично ФНФ типа статическое ОЗУ. Генерируемое выход-

ное значение при этом можно использовать для получения уникального неклонированного идентификатора.

2. *Режим кольцевого генератора.* В данном режиме схема функционирует при удержании  $EN = 1$ , при этом в цепи обратной связи, соединяющей выход схемы с единичным входом мультиплексора, включено нечетное количество инверторов, а на выходе  $Q$  вырабатывается сигнал в виде меандра с уникальной частотой, обусловленной структурными особенностями всех элементов схемы и их проводящих линий.
3. *Режим хранения.* В этот режим схема переходит при переключении управляющего сигнала  $EN$  из 1 в 0. При этом в цепи обратной связи конфигурируется четное число инверторов и схема становится эквивалентной схеме бистабильного элемента.

## II. РЕАЛИЗАЦИЯ ФНФ НА FPGA

Реализация произвольной комбинационной схемы на FPGA осуществляется при помощи LUT-блоков. Предложенная схема ФНФ в минимальной ее конфигурации ( $k = 0$ ) может быть реализована на одном LUT-блоке. При этом, для уменьшения аппаратных затрат, функция инвертора может быть перенесена в конфигурацию LUT. Современные FPGA, такие как Xilinx Artix-7 имеют в своем составе 6-входовые LUT-блоки, что может быть использовано для расширения реализации предложенной ФНФ. На рисунке 2 приведены схемы упаковки ФНФ в технологические блоки LUT-3 и LUT-6.

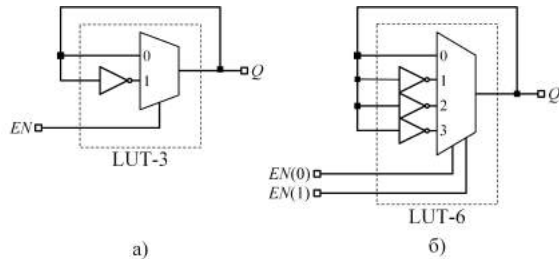


Рис. 2 – Схема упаковки ФНФ на блоках LUT-3 (а) и LUT-6 (б)

В модифицированной схеме ФНФ на LUT-6 вместо сигнала  $EN$  используются два селективных сигнала  $EN(0)$  и  $EN(1)$ , определяющие следующие режимы функционирования:  $EN = 00$  – режим бистабильного элемента,  $EN = \{01, 10, 11\}$  – режимы трех принципиально различных кольцевых генераторов, вырабатывающих сигналы с различными частотами в зависимости от значения  $EN$ :  $F_Q^{01} \neq F_Q^{10} \neq F_Q^{11}$ .

Для реализации данной схемы на LUT-6 была использована FPGA Xilinx Artix-7 XC7A100T, входящая в состав платы быстрого прототипирования Digilent NEXYS 4.

## III. АНАЛИЗ ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ

Для проведения экспериментов были собраны 16 схем ФНФ на LUT-6 блоках с общим управлением по входным линиям  $EN$  от устройства управления на основе софт-процессора MicroBlaze и UART-компоненты, посредством которой данные передавались для анализа от платы NEXYS 4 на рабочую станцию.

В ходе проведенных экспериментов было установлено, что только две из 16 ФНФ-схем (схемы с индексом 2 и 3) вырабатывают нестабильные значения в режиме инициализации. Вероятности появления единичного символа для этих схем следующие:  $P_1^2 = 0,10$  и  $P_1^3 = 0,49$ , что свидетельствует о возможном применении данной схемы ФНФ для построения неклонированных идентификаторов цифровых устройств.

На рисунке 3 приведены значения вероятностей появления единичного символа в трех режимах кольцевых генераторов для всех 16 схем.

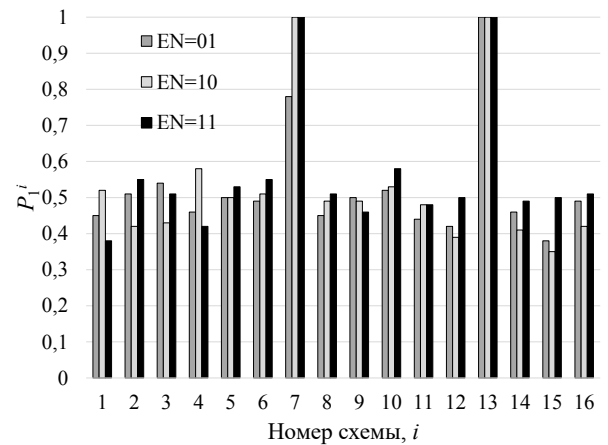


Рис. 3 – Значения вероятностей  $P_1^i$  для трех режимов схемы ФНФ

## IV. ЗАКЛЮЧЕНИЕ

В работе предложена компактная схемотехническая реализация ФНФ, использующая ресурсы одного блока LUT-6 кристалла FPGA Xilinx Artix-7. Планируются дальнейшие исследования характеристик предложенной ФНФ и разработка схем генерирования уникальных идентификаторов и случайных числовых последовательностей на ее основе.

## V. СПИСОК ЛИТЕРАТУРЫ

1. Suh, G. E. Physical unclonable functions for device authentication and secret key generation / G. E. Suh, S. Devadas // ACM/IEEE Design Automat. Conf. (DAC'2007). – San Diego, USA, 2007. – P. 9–14.
2. The bistable ring PUF: A new architecture for strong physical unclonable functions / Q. Chen [et al.] // Proc. IEEE Int. Sympos. on Hardw. Orient. Secur. and Trust (HOST'11). – San Diego, USA, 2011. –P. 134–141.

# О ФУНКЦИОНАЛЬНЫХ РАЗЛОЖЕНИЯХ СВЯЗАННЫХ СИСТЕМ БУЛЕВЫХ ФУНКЦИЙ

Кардаш С. Н.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: kardash77@gmail.com

*Рассматривается задача построения совместных (использующих общие подфункции) дизъюнктивных и конъюнктивных разложений систем булевых функций. Приводятся результаты экспериментального исследования компьютерных программ, реализующих разработанные алгоритмы построения таких разложений.*

## I. ВВЕДЕНИЕ

В настоящее время стремление уменьшить площадь цифровых систем, реализуемых на элементной базе заказных комплементарных металл-оксид-полупроводниковых схем (КМОП-схем) и систем-на-кристалле стало причиной появления новых и совершенствования известных методов решения задач, связанных с проектированием логических схем. Синтез логических схем из библиотечных элементов обычно выполняется по оптимизированным двухуровневым либо многоуровневым представлениям систем булевых функций. Двухуровневыми (И-ИЛИ) представлениями называют представления функций в виде дизъюнктивных нормальных форм (ДНФ), многоуровневыми – различные формы функциональных разложений [1,2]. Идея использовать связанность (общность) областей определений булевых функций при синтезе многовыходных комбинационных схем предложена в [3].

«Хорошая» связанность функций существенно влияет на появление одинаковых структурных частей (конъюнкций, алгебраических выражений, подфункций и т.д.) в оптимизированных двухуровневых либо многоуровневых формах представления функций, по которым и строятся логические схемы в том или ином технологическом базисе. Чем сильнее связаны функции, тем скорее можно ожидать, что в представлениях таких функций будет больше одинаковых подвыражений и синтезированные схемы будут менее сложными. По существу, выделение связанных функций является одним из приемов логической оптимизации многоуровневых представлений систем функций [4]. Для связанных подсистем функций более эффективно решаются задачи логической оптимизации, например, оптимизации в классе ДНФ [2, 3], оптимизации BDD-представлений [4] и декомпозиции различных видов, например, при построении совместных функциональных разложений [5]. В данной работе для многоуровневой оптимизации систем функций, обладающих связанностью областей определения, предлагается использовать совместные функциональные разложения и приводятся результаты эксперименталь-

ного исследования программ построения таких разложений.

## II. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ И ПОСТАНОВКА ЗАДАЧИ.

Представим исходную систему СДНФ в матричном виде парой булевых матриц –  $U$  и  $S$ . Столбцы матрицы  $U$  соответствуют аргументам системы, а строки задают элементарные конъюнкции. Столбцы матрицы  $S$  соответствуют функциям системы, а единичные значения элементов в матрице  $S$  отмечают вхождения соответствующих конъюнкций в ДНФ функций системы. Будем считать, что функции системы доступны как в прямом, так и в инверсном виде. В этом случае, очевидно, для любой системы СДНФ всегда существует дизъюнктивное разложение – т.е. всегда найдется компонента связности  $h(x)$  мощности не ниже 1. Остаточные функции будут индивидуальными для каждой функции исходной системы. Их удобно объединить в одну систему  $g(x)$ . Назовем такое представление системы ее дизъюнктивным разложением. Очевидно также, что для решения практических задач имеет смысл искать компоненты связности максимальной, по крайней мере, больше единичной мощности.

Постановка задачи. Пусть задана система СДНФ булевых функций. Требуется выделить в ней  $w$ -связную (заданную на  $w$  элементарных конъюнкциях) подсистему, содержащую не менее  $k$  ( $k > 1$ ) функций. Для полученной подсистемы построить дизъюнктивное разложение. Из функций, не вошедших в подсистему, образовать подсистему «остаток».

В качестве критерия для оценки качества получаемого разложения будем использовать суммарную площадь (вес) представляющих разложение матриц. От этой величины напрямую зависит площадь схемы, при реализации ее на кристалле.

Для решения этой задачи предлагается эвристический алгоритм. Идея предлагаемого алгоритма состоит в пошаговом формировании  $w$ -связной подсистемы путем выбора подходящих функций с учетом их полярности и последую-

щего включения их в подсистему. Числа  $w$  и  $k$  служат параметрами алгоритма.

### III. ОРГАНИЗАЦИЯ ЭКСПЕРИМЕНТА И РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНОГО ИССЛЕДОВАНИЯ

Предложенные алгоритмы реализованы в виде программ на языке C++. В настоящей работе приводятся результаты экспериментального исследования эффекта от применения программы, разработанной на основе алгоритма построения дизъюнктивного разложения. Исследование проводилось в системе FLC [6]. В качестве объектов исследования использовались системы полностью определенных булевых функций, взятые из библиотеки примеров Berkeley PLA Test Set [7]. Для каждого примера применялась многошаговая процедура, состоящая в циклическом построении дизъюнктивного разложения. Параметры разложения во всех случаях имели одинаковые значения:  $w = 5$ ,  $k = 2$ . В результате получалось многоблочное представление схемы на языке SF. При этом оценивалась площадь получаемого схемного решения.

Результаты экспериментального исследования представлены в таблице 1. Здесь:  $n$  – число переменных,  $m$  – число функций,  $l$  – число элементарных конъюнкций исходной системы СДНФ булевых функций. Результат:  $A$  – площадь исходной СДНФ, представленной в матричном виде,  $B$  – площадь разложения, PROC – отношение результата к исходному заданию.

### IV. ЗАКЛЮЧЕНИЕ

Уменьшить площадь получаемых в процессе синтеза комбинационных нерегулярных логических схем можно с помощью логической оптимизации исходных описаний. Одним из наиболее эффективных способов логической оптимизации

является минимизация BDD-представлений систем булевых функций. Хорошим средством для улучшения конечных схемных решений может служить предварительная обработка исходных систем с помощью программ построения дизъюнктивно-конъюнктивных разложений. Проведенное исследование программ, разработанных на основе предложенных алгоритмов, показало, что во всех исследованных случаях их применение приводило к сокращению площади получаемых схемных решений. При этом использование разработанных программных средств может сократить площадь получаемых схем на 25 процентов.

### V. СПИСОК ЛИТЕРАТУРЫ

1. Закревский, А. Д. Логические основы проектирования дискретных устройств / А. Д. Закревский, Ю. В. Поттосин, Л. Д. Черемисинова. – М.: Физматлит, 2007. – 592 с.
2. Бибило, П. Н. Декомпозиция булевых функций на основе решения логических уравнений / П. Н. Бибило. – Минск: Беларус. навука, 2009. – 211 с.
3. Кузнецов, О. П. О программной реализации логических функций и автоматов / О. П. Кузнецов // Автоматика и телемеханика. – 1977. – № 7. – С. 63–74.
4. Akers, S. B. Binary decision diagrams / S. B. Akers // IEEE Trans. on Computers. – 1978. – Vol. C-27, no. 6. – P. 509–516.
5. Бибило, П.Н. Дизъюнктивно-конъюнктивные разложения систем полностью определенных булевых функций / П.Н. Бибило, С.Н. Кардаш, / Доклады Восьмой Международной научной конференции «Танаевские чтения», 27–30 марта 2018 г. – Минск, ОИПИ НАН Беларуси, 2018 г. – С. 28–32.
6. Бибило П.Н., Романов В.И. Логическое проектирование дискретных устройств с использованием продукционно-фреймовой модели представления знаний. Изд. 2-е, испр. – М.: ЛЕНАНД, 2014, 256 с.
7. Berkeley PLA test set [Electronic resource]. – Mode of access: <http://www1.cs.columbia.edu/cs4861/sis/espresso-examples/ex/>. – Date of access: 10.10.2011.

Таблица 1 – Результаты эксперимента

| Имя     | $n$ | $m$ | $l$     | $A$    | $B$    | PROC |
|---------|-----|-----|---------|--------|--------|------|
| add6    | 9   | 8   | 512     | 77824  | 76243  | 98   |
| addm4   | 9   | 8   | 512     | 8704   | 8480   | 97   |
| max512  | 9   | 6   | 512     | 7680   | 7535   | 98   |
| max1024 | 10  | 6   | 1024    | 16384  | 16094  | 98   |
| mp2d    | 14  | 14  | 16384   | 458752 | 335200 | 73   |
| tial    | 14  | 8   | 16384   | 360448 | 356752 | 99   |
| dk48    | 15  | 17  | 1048576 | 524736 | 356752 | 50   |
| m4      | 8   | 16  | 256     | 6144   | 4914   | 80   |
| m3      | 7   | 16  | 128     | 2944   | 2449   | 83   |
| z5xp    | 7   | 10  | 128     | 2176   | 2167   | 99   |
| root    | 8   | 5   | 256     | 3328   | 3013   | 90   |
| adr4    | 8   | 5   | 2560    | 3328   | 3069   | 92   |

# ВЛИЯНИЕ ДЛИТЕЛЬНОСТИ РАБОТЫ КОЛЬЦЕВОГО ОСЦИЛЛЯТОРА НА СТАТИСТИЧЕСКИЕ ХАРАКТЕРИСТИКИ ПОСЛЕДОВАТЕЛЬНОСТИ БИТ, СГЕНЕРИРОВАННОЙ АППАРАТНЫМ ГЕНЕРАТОРОМ СЛУЧАЙНЫХ ЧИСЕЛ

Кохновский С. И., Иванюк А. А

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь

E-mail: stan.ver.i.esk.slav@gmail.com, ivaniuk@bsuir.by

В данной работе описано влияние длительности работы кольцевого осциллятора на статистические характеристики последовательности бит, сгенерированной аппаратным генератором случайных чисел реализованным на плате FPGA Digilent Nexys 4.

## ВВЕДЕНИЕ

В настоящее время существует достаточно широкий спектр аппаратных генераторов случайных чисел. Однако несмотря на высокие статистические показатели, известные генераторы обладают и существенным недостатком – высокой аппаратной сложностью. Данный недостаток критичен при применении генераторов в проектах с значительными ограничениями в ресурсах. Решением проблемы является использование относительно простой системы, удовлетворяющей заданному порогу качества.

## I. ОПИСАНИЕ СТРУКТУРЫ ГЕНЕРАТОРА СЛУЧАЙНЫХ ЧИСЕЛ

Архитектурно компонент реализован на основе  $D$ -триггера, где бит данных ( $D$ ) соединён с битом выхода ( $Q$ ) линией с инвертором. Данная конфигурация, представляющая собой комбинацию триггера и кольцевого осциллятора, может использоваться как аппаратный генератор случайных чисел, так как наличие кольцевого осциллятора способствует вводу триггера в метастабильное состояние (см. рис. 1).

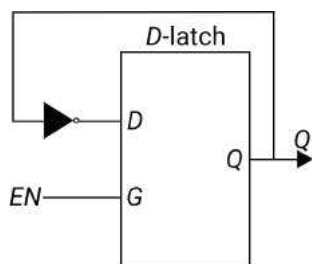


Рис. 1 – Логический компонент аппаратного генератора случайных чисел

Однако вход системы в метастабильное состояние занимает некоторое время после установки сигнала  $EN$  в логическую единицу, а сгенерированные значения не всегда являются статистически случайными из-за влияния большого количества иных факторов (смещения фазы, всевозможных задержек, аппаратных особенностей реализации FPGA и др.). В данной работе

рассмотрено влияние продолжительности подачи сигнала  $EN$  триггеру на характеристики сгенерированной последовательности бит.

Для более наглядной демонстрации поведения системы реализован IP-компонент, состоящий из  $num\_gen = 32$  описанных выше генераторов случайных чисел, а также содержащий логику управления длительностью передачи сигнала  $EN$  на вход генераторов, причём длительность передачи кратна длительности периода сигнала синхронизации системы. Сигнал  $EN$  подаётся на вход всем генераторам чисел в течение равного количества периодов синхронизации. В качестве управляющего компонента выбран процессор microblaze, сконфигурированный для работы на частоте 100 МГц.

## II. ОПИСАНИЕ ЭКСПЕРИМЕНТА

Эксперимент состоит в получении и исследовании характеристик сгенерированных последовательностей. Длины сгенерированных каждым из  $num\_gen$  генераторов последовательностей составили 1000 бит. Обозначим количество единиц, сгенерированное генератором под номером  $i$ , как  $num\_ones_i$ , тогда количество единиц, сгенерированное всеми генераторами, равно  $num\_ones = \sum_{i=1}^{num\_gen} num\_ones_i$ . Аналогично, для общего количества сгенерированных чисел используем обозначение  $len = \sum_{i=1}^{num\_gen} len_i$ , где количество сгенерированных символов генератором  $i$  обозначим  $len_i$ . Также введём обозначение для среднего арифметического сгенерированных символов:  $mean = \frac{num\_ones}{len}$ . При продолжительности установки бита  $EN = 1$  в  $k = 1$  период сигнала синхронизации большая часть генераторов возвращает одно и то же значение - 0 (см. рис. 2). На рисунке зелёным цветом выделен график  $y = 0.5$  (половина сгенерированных символов - 1). Красным цветом обозначен график  $y = mean$ . Из информации на графике можно сделать вывод, что процент единиц от общего количества сгенерированных символов не достигает 12%.

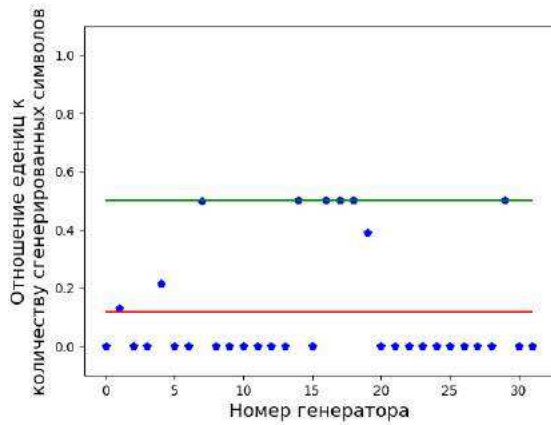


Рис. 2 – Отношение количества единиц к количеству сгенерированных символов для  $num\_gen$  генераторов при подаче сигнала  $EN$  в течение  $k = 1$  периода сигнала синхронизации

При увеличении времени удержания сигнала  $EN$  в единице до 8 периодов сигнала синхронизации увеличился разброс измерений, а значение  $mean$  приблизилось к 0.5 (см. рис. 3), что свидетельствует о более случайной природе работы генераторов, впрочем, о статистической случайности речи не идёт -  $num\_ones$  остаётся на отметке менее 34% от количества всех сгенерированных символов, а половина генераторов генерируют последовательности с сильным преобладанием нулей или единиц.

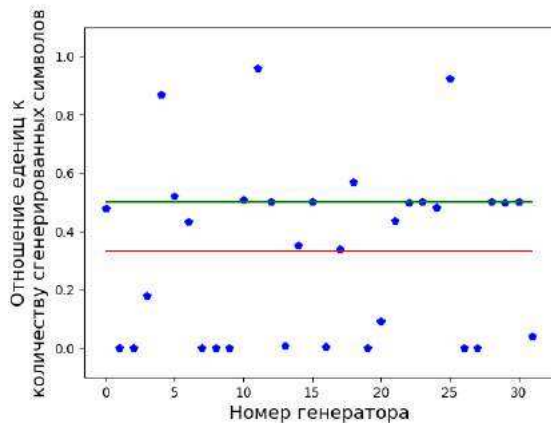


Рис. 3 – Отношение количества единиц к количеству сгенерированных символов для  $num\_gen$  генераторов при подаче сигнала  $EN$  в течение  $k = 8$  периодов сигнала синхронизации

При дальнейшем увеличении времени подачи сигнала  $EN$  до 1024 циклов сигнала синхронизации соотношение  $mean$  стало ближе к 0.5, в то время как на всём множестве генераторов наблюдается выравнивание значений  $num\_ones_i$  за исключением редких выбросов, частота появления которых также уменьшается по мере увеличения времени генерации (см. рис. 4).

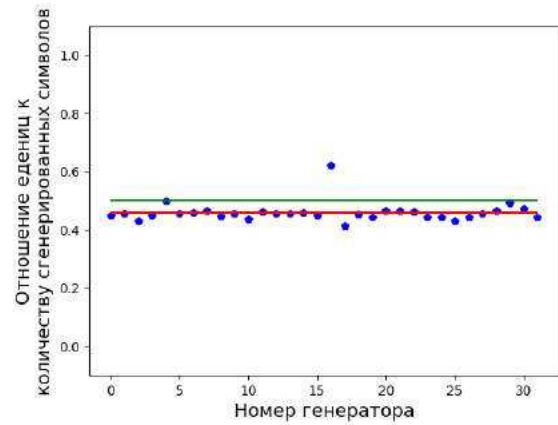


Рис. 4 – Отношение количества единиц к количеству сгенерированных символов для  $num\_gen$  генераторов при подаче сигнала  $EN$  в течение  $k = 1024$  периодов сигнала синхронизации

Рассмотрим диапазоны, в которые попадают измерения на всех генераторах с увеличением времени генерации (таблица 1). Из-за высокого количества нулей при  $k < 8$  дальность разброса измерений относительно невысока (около 0.5), однако после  $k = 4$  диапазон значений становится близок к единице, но с значения  $k = 8$  начинает убывать.

Таблица 1 – Начало и ширина диапазона, в который попадают значения  $\frac{num\_ones_i}{len_i}, i = \overline{1, num\_gen}$

|       | $k$ |      |       |       |       |       |
|-------|-----|------|-------|-------|-------|-------|
|       | 1   | 8    | 64    | 256   | 1024  | 65536 |
| от    | 0.0 | 0.0  | 0.2   | 0.427 | 0.415 | 0.405 |
| диап. | 0.5 | 0.96 | 0.449 | 0.15  | 0.205 | 0.086 |

## Выводы

Проведённое исследование позволяет сделать следующие выводы:

- при увеличении продолжительности подачи сигнала  $EN = 1$  на вход генератора числовой последовательности, улучшаются статистические показатели сгенерированной последовательности, такие как приближение доли единиц к 0.5, выравнивание доли нулей и единиц в блоках произвольной длины, а также длительностей монотонного генерирования 0 или 1;
- наиболее быстрое выравнивание количества единиц в каждой из сгенерированных последовательностей происходит быстрее всего после  $k = 8$ , но при приближении к  $k = 256$  улучшение показателей заметно не столь ярко.

## СПИСОК ЛИТЕРАТУРЫ

1. An Area Efficient True Random Number Generator Based on Modified Ring Oscillators [Electronic resource] / Mehmet Alp Şarkışla, Salih Ergün. – Institute of Electrical and Electronics Engineers, 2018. – Mode of access: <https://ieeexplore.ieee.org/document/8605697>. – Date of access: 20.10.2021.

# АНАЛИЗ ТЕСТОВ ЗАПОМИНАЮЩИХ УСТРОЙСТВ ДЛЯ ОБНАРУЖЕНИЯ ПАССИВНЫХ КОДОЧУВСТВИТЕЛЬНЫХ НЕИСПРАВНОСТЕЙ

Петровская В. В., Деменковец Д. В.

Кафедра программного обеспечения информационных технологий,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь

E-mail: vita.petrovskaya@gmail.com, Demenkovets@bsuir.by

*Тезис посвящен методам обнаружения неисправностей запоминающих элементов. Рассматриваются модели неисправностей, структура и алгоритм выполнения маршевых тестов. Отражается принцип выполнения неразрушающих версий маршевых тестов, приводятся результаты обнаружения пассивных граничных кодочувствительных неисправностей неразрушающими тестами на базе двойных адресных последовательностей.*

## ВВЕДЕНИЕ

В вычислительных машинах значительную роль играет система памяти. Крайне важно сохранить целостность хранимых на устройстве данных. Причиной неисправного состояния памяти является физический или механический дефект. Выделяют два подмножества неисправностей: неисправности матрицы запоминающих устройств и неисправности электронного обрамления (дешифраторы адреса, логика записи/чтения). К первому подмножеству относится обширное разнообразие неисправностей, в которых участвует одна, две или несколько ячеек памяти [1, 2].

Неисправные состояния запоминающих устройств описываются математическими моделями, абстракциями ошибок, вызванными конкретными физическими дефектами [2]. Среди неисправностей, в которых участвуют несколько ячеек запоминающего устройства, выделяют кодочувствительные неисправности (Pattern Sensitive Fault – PSF). В неисправности PSF могут участвовать любые  $k$  из  $N$  ячеек запоминающего устройства, одна из них зависимая (Base Cell), остальные соседние (Neighborhood Cells). Для тестирования значение  $k$  не превышает 10, поскольку необходимое для теста время пропорционально  $2^k$ . Такая модель называется граничная кодочувствительная неисправность (Neighborhood Pattern Sensitive Fault – NPSF) [2].

Тестирование полупроводниковой памяти прошло долгий процесс эволюции. В начале 1980-х годов, после внедрения моделей неисправностей, доминирующим видом тестирования стали маршевые тесты. Они показывают хорошие результаты покрытия неисправностей и обычно линейно зависят от объема памяти, что делает их приемлемыми для промышленной реализации [3].

Маршевый тест состоит из конечной последовательности маршевых элементов, записанных в фигурных скобках. Маршевый элемент (фа-

за) представляет собой конечную последовательность операций чтения и/или записи, которая применяется к каждой ячейке памяти. Операции в маршевом элементе имеют следующие обозначения: «r0» – считывание 0 из ячейки памяти, «r1» – считывание 1 из ячейки памяти, «w0» – запись 0 в ячейку памяти, «w1» – запись 1 в ячейку памяти. В начале каждого элемента указывается порядок адресов тестируемых ячеек: по возрастанию ( $\uparrow$ ), по убыванию ( $\downarrow$ ), по возрастанию или убыванию ( $\downarrow\uparrow$ ) [1, 3].

Важной проблемой сложных систем с высокими требованиями к надежности является возможность проведения тестирования в режиме онлайн (без прерывания) или с минимальным прерыванием нормальной работы всей системы. Для такого тестирования содержимое памяти не должно быть изменено [1, 4].

## I. АНАЛИЗ НЕРАЗРУШАЮЩИХ ТЕСТОВ ЗУ

В процессе исследований в области тестирования памяти была предложена последовательность шагов по преобразованию маршевого теста в его неразрушающую версию [5]. Для проведения неразрушающего тестирования используется буфер, который хранит значение текущей ячейки памяти. Неразрушающие операции чтения записывают в буфер значение в прямом (rd) или инверсном виде (rd\*), а неразрушающие операции записи записывают прямое (wd) или обратное (wd\*) значение из буфера в текущую ячейку памяти. Для того чтобы определить, есть ли дефекты в памяти, сравнивают эталонную и реальную сигнатуру, полученные в процессе выполнения начального и базового тестов. Сигнатуры получают сжатием прочитанных данных на сигнатурном анализаторе.

Сравнение сигнатур позволяет определить, исправна ли матрица запоминающих элементов, но не дает информацию об активизированных неисправностях. Новый метод неразрушающего тестирования, основанный на базе двойных адресных последовательностей, обеспечивает и ак-



тивизацию, и обнаружение неисправностей. В основе таких тестов лежит базовый элемент  $2\uparrow(\text{rd}, \text{wd}^*, \text{rd})$ . Операция записи противоположного значения активизирует неисправность, а вторая операция чтения обнаруживает ее. В случае исправного состояния ячейки памяти двукратное применение базового элемента восстанавливает исходное значение, поэтому для проверки наличия неисправности в памяти достаточно сравнить сигнатуру первоначального содержимого с сигнатурой конечных данных. Описание тестов с базовыми элементами представлены в таблице 1 [6].

Таблица 1 – Неразрушающие маршевые тесты на базе двойных адресных последовательностей

| Название теста | Описание теста                                                                                                                                               | Сложность теста |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|
| March_2A_1     | $\uparrow\downarrow(\text{rd}); 2\uparrow(\text{rd}, \text{wd}^*, \text{rd}); \uparrow\downarrow(\text{rd})$                                                 | 8N              |
| March_2A_2     | $\uparrow\downarrow(\text{rd}); 2\uparrow(\text{rd}, \text{wd}^*, \text{rd}); 2\downarrow(\text{rd}, \text{wd}^*, \text{rd}); \uparrow\downarrow(\text{rd})$ | 14N             |

Двойные адресные последовательности для покрытия N ячеек памяти формируются на базе последовательности 2N адресов. Задается набор индексов, указывающий порядок и номера битов, участвующих в формировании адреса в новой двойной последовательности. Пример изменения содержимого 8-битного массива памяти при выполнении теста March\_2A\_1 представлен в таблице 2.

Таблица 2 – Пример выполнения теста March\_2A\_1

| $A_c = c_3c_2c_1c_0$ | $2A_c = c_3c_1c_0$ | 11111010 |
|----------------------|--------------------|----------|
| 0000                 | 000                | 11111011 |
| 0001                 | 001                | 11111001 |
| 0010                 | 010                | 11111101 |
| 0011                 | 011                | 11110101 |
| 0100                 | 000                | 11110100 |
| 0101                 | 001                | 11110110 |
| 0110                 | 010                | 11110010 |
| 0111                 | 011                | 11111010 |
| 1000                 | 100                | 11101010 |
| 1001                 | 101                | 11001010 |
| 1010                 | 110                | 10001010 |
| 1011                 | 111                | 00001010 |
| 1100                 | 100                | 00011010 |
| 1101                 | 101                | 00111010 |
| 1110                 | 110                | 01111010 |
| 1111                 | 111                | 11111010 |

Выполнение теста March\_2A\_1 с двойной адресной последовательностью  $2A_c = c_3c_1c_0$  и исходным содержимым памяти 11111010 позволяет обнаружить 6 неисправностей PNPSF3 в ячейках с адресами 0, 1, 2:  $\langle 0,1,\uparrow \rangle$ ,  $\langle 1,0,\downarrow \rangle$ ,  $\langle 1,\uparrow,0 \rangle$ ,  $\langle 0,\downarrow,1 \rangle$ ,  $\langle \uparrow,0,1 \rangle$ ,  $\langle \downarrow,1,0 \rangle$ .

В ходе исследования была написана программа, моделирующая пассивные граничные кодочувствительные неисправности памяти

(PNPSFk) и реализующая однократное и многократное неразрушающее тестирование на базе двойных адресных последовательностей. Результаты тестирования 64-битного массива памяти с произвольным содержимым отражены в таблицах 3,4. Первое значение получено однократным выполнением теста с адресной последовательностью  $2A_c = c_6c_5c_4c_3c_1c_0$ , полученной из счетчиковой последовательности. Второе значение показывает суммарную полноту покрытия в двух последовательных итерациях. Во второй итерации применяется адресная последовательность  $2A_g = g_6g_5g_4g_3g_1g_0$ , полученная на основании последовательности Грея.

Таблица 3 – Полнота покрытия PNPSFk тестом March\_2A\_1

| Номер итерации | PNPSF3  | PNPSF5 | PNPSF7 | PNPSF9 |
|----------------|---------|--------|--------|--------|
| 1              | 25%     | 6,25%  | 1,563% | 0,391% |
| 2              | 37,634% | 10%    | 2,57%  | 0,651% |

Таблица 4 – Полнота покрытия PNPSFk тестом M\_2A\_2

| Номер итерации | PNPSF3  | PNPSF5  | PNPSF7 | PNPSF9 |
|----------------|---------|---------|--------|--------|
| 1              | 45,968% | 11,875% | 3,017% | 0,76%  |
| 2              | 50,269% | 13,75%  | 3,579% | 0,911% |

## II. ЗАКЛЮЧЕНИЕ

Неразрушающие тесты на базе двойных адресных последовательностей решают проблему диагностирования неисправностей и существенно сокращают время тестирования за счет двукратного инвертирования содержимого памяти. Покрывающая способность тестов зависит от выбранной адресной последовательности. Многократное использование тестов позволяет заметно увеличить полноту покрытия неисправностей.

## III. СПИСОК ЛИТЕРАТУРЫ

1. Ярмолик, В. Н. Маршевые тесты для самотестирования ОЗУ / В. Н. Ярмолик, А. П. Запкович, А. А. Иванов // Минск: Бестпринт. – 2009.
2. Bushnell, M. L. Essentials of Electronic Testing for Digital, Memory & Mixed-Signal VLSI Circuits / M. L. Bushnell // New York: Kluwer Academic Publishers. – 2000.
3. Hamdioui S. Testing Embedded Memories: A Survey Computer Engineering / S. Hamdioui // Mathematical and Engineering Methods in Computer Science. – 2013. – Vol. 7721. – P. 32-42.
4. Demidenko, S. N. Simulation and development of short transparent tests for RAM / S. N. Demidenko // Proceedings 10th Asian Test Symposium. – 2001. – P. 164-169.
5. Mrozek I. MATS+ transparent memory test for Pattern Sensitive Fault detection / I. Mrozek, V. N. Yarmolik // 15th International Conference on Mixed Design of Integrated Circuits and Systems. – 2008. – P. 493-498.
6. Ярмолик, В. Н. Неразрушающее тестирование запоминающих устройств на базе двойных адресных последовательностей / В. Н. Ярмолик, И. Мрозек, В. А. Леванцевич, Д. В. Деменковец // Доклады БГУИР. – 2021. – № 19(4). – С. 43-51.

# АНАЛИЗ ХАРАКТЕРИСТИК ФИЗИЧЕСКИ НЕКЛОНИРУЕМЫХ ФУНКЦИЙ ТИПА АРБИТР РАЗЛИЧНЫХ КОНФИГУРАЦИЙ

Шамына А. Ю., Иванюк А. А.

Кафедра программного обеспечения информационных технологий, кафедра информатики,  
Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {shamyna, ivaniuk}@bsuir.by

*Произведен анализ характеристик случайности, уникальности и стабильности физически неклонированных функций типа арбитр различных конфигураций. Кратко описано построение экспериментальной установки для исследования. Показана зависимость характеристик от длин блока симметричных путей АФНФ, а также выбранного арбитра.*

## ВВЕДЕНИЕ

Физически неклонированные функции (ФНФ) имеют широкое применение в физической криптографии. Реализованные ФНФ на FPGA привлекательны благодаря относительной простоте реализации и небольшим аппаратным затратам. Для изучения выбраны ФНФ типа арбитр (АФНФ) [1], реализованные на FPGA Artix 7 фирмы Xilinx, входящих в комплектацию плат быстрого прототипирования Digilent Nexys 4 [2]. Принцип, на котором основывается работа АФНФ, заключается в извлечении производственной энтропии при прохождении тестовых импульсов через звенья АФНФ, выражающейся в различных временных задержках для каждого экземпляра АФНФ. Обычно в классической структуре АФНФ выделяют генератор тестовых сигналов (ГТС), блок симметричных путей (БСП), а также арбитр, отвечающий за выработку ответа ФНФ.

## I. ПОДГОТОВКА ЭКСПЕРИМЕНТА И СБОР ДАННЫХ

Для проведения эксперимента на языке VHDL было создано проектное описание экспериментальной установки изучения АФНФ с использованием САПР Vivado 2016.4. Проект включает в себя ГТС, устройство управления на основе цифрового конечного автомата (ЦКА), а также аппаратный генератор M-последовательности в виде сдвигового регистра с линейной обратной связью (linear feedback shift register, LFSR). Передача данных между установкой и ПК организована через интерфейс UART (Universal Asynchronous Receiver-Transmitter). Поддержка передачи данных через UART со стороны экспериментальной установки реализована с использованием IP ядер и софтверного процессора Microblaze.

Исследование проводилось для АФНФ различных конфигураций: с различным числом звеньев БСП  $N$  ( $N = 16$ ,  $N = 32$ ,  $N = 64$  и  $N = 128$ ), а также с арбитрами на базе RS-защелки и D-триггера.

Для определения характеристик АФНФ для всех конфигураций было проведено  $E = 10$  экспериментов на  $M = 10$  устройствах. Каждый эксперимент включал генерацию  $C = 10^6$  запросов к ФНФ и сбор такого же количества ответов. Также для вычисления внутрикристалльной уникальности было реализовано  $D = 16$  идентичных АФНФ на каждом устройстве.

## II. ИССЛЕДОВАНИЕ ХАРАКТЕРИСТИК АФНФ

Одной из ключевых характеристик ФНФ является стабильность, которая определяет стабильность ответа на фиксированный запрос для конкретного экземпляра ФНФ. Особенно актуальна эта характеристика, например, при использовании ФНФ в качестве средства генерации аппаратного идентификатора. Для классических реализаций АФНФ значение характеристики стабильности может быть снижено из-за особенностей выбранного арбитра, а также изменений внешних условий. Формально значение стабильности  $S(CH)$  ответа ФНФ  $R$  на запрос  $CH$  определяется следующим образом (формула 1):

$$S(CH) = 1 - \frac{1}{E} \sum_{e=1}^E HD(R_{ref}, R_e), \quad (1)$$

где  $E$  – количество экспериментов;  $e$  – индекс эксперимента;  $HD$  – расстояние Хемминга;  $R_{ref}$  – эталонное значение ответа на заданный запрос, определяемое по мажоритарному принципу;  $R_e$  – ответ на заданный запрос.

Стабильность каждой конфигурации АФНФ определялась как среднее арифметическое всех значений стабильности выполненных запросов и представляется формулой (2):

$$\frac{1}{K} \sum_{i=1}^k S(CH_i) \quad (2)$$

где  $K$  – количество запросов,  $i$  – индекс запроса.

Полученные результаты характеристик стабильности представлены на рисунке 2.

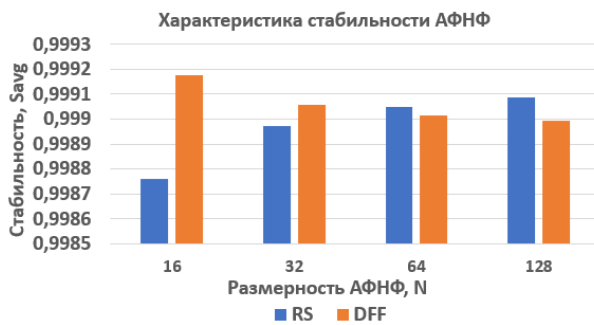


Рис. 1 – Зависимость значения средней стабильности АФНФ от размерности и выбранного арбитра

Метрика уникальности АФНФ для реализаций на идентичных интегральных схемах [3], а также копий АФНФ на одном устройстве, представляет собой усредненное расстояние Хэмминга между экземплярами ФНФ и рассчитывается согласно формуле:

$$U(PUF) = \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^m \frac{HD(R_u, R_v)}{l},$$

где  $m$  - количество экземпляров ФНФ,  $l$  - разрядность ответа ФНФ, бит.

Характеристики внутрикристалльной и межкристалльной уникальности, полученные в результате эксперимента, представлены на рисунках 2, 3.

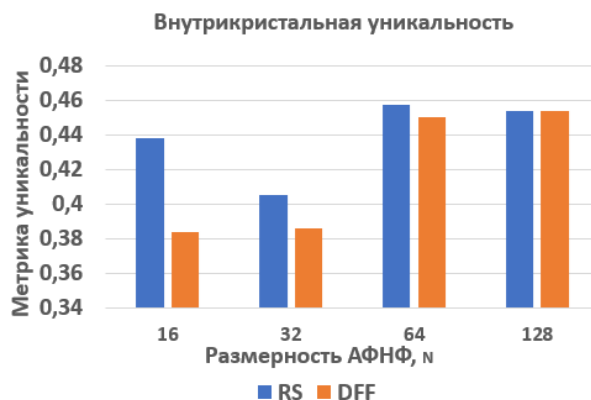


Рис. 2 – Зависимость значения внутрикристалльной уникальности АФНФ от размерности и выбранного арбитра

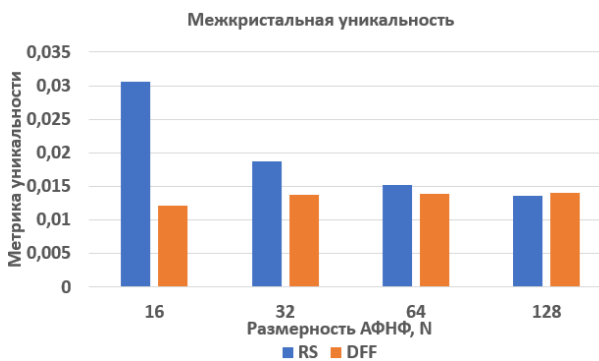


Рис. 3 – Зависимость значения межкристалльной уникальности АФНФ от размерности и выбранного арбитра

Для оценки случайности была взята частота появления ответа  $r = 1$ . Пусть с использованием ФНФ сгенерирована последовательность ответов  $R$  длиной  $n$ , тогда вероятность появления символа  $\alpha$   $p_\alpha$ , встретившегося в  $R$  равно  $k_\alpha$ , определяется через отношение:

$$p_\alpha = \frac{k_\alpha}{n}.$$

Экспериментальные результаты  $p_1$  отображены в виде графика на рисунке 4.

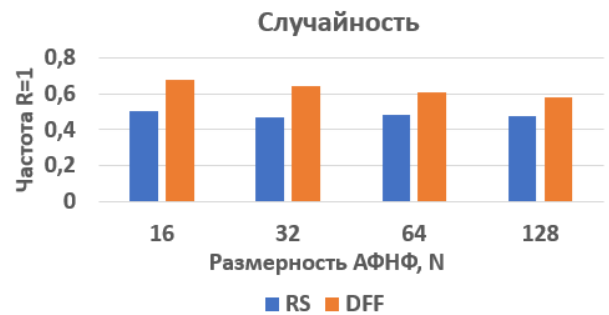


Рис. 4 – Зависимость значения вероятности ответа АФНФ  $r = 1$  от размерности и выбранного арбитра

Полученные результаты свидетельствуют о приближении значения  $p_1$  к эталонному  $p_1 = 0.5$  при увеличении количества звеньев БСП.

### III. ЗАКЛЮЧЕНИЕ

На основе полученных результатов можно сделать вывод о преимущественно улучшении характеристик ФНФ с увеличением размерности АФНФ. Несколько лучшие характеристики демонстрируют реализации АФНФ с использованием RS-защелки в качестве арбитра. Следует отметить, что при увеличении размерности АФНФ возрастают аппаратные затраты и время генерации ответа.

Существенным недостатком исследованных реализаций АФНФ являются низкие значения межкристалльной уникальности. Это может сильно затруднить их использование и потребовать использования дополнительных модификаций.

### IV. СПИСОК ЛИТЕРАТУРЫ

1. A technique to build a secret key in integrated circuits for identification and authentication applications / J.W. Lee [et al.] // Intern. Symp. VLSI Circuits (VLSI'04), Honolulu, USA, June 15–19, 2004. – Honolulu, 2004. – P. 176–179
2. Nexys 4 artix-7 FPGA: Trainer board recommended for ece curriculum [Electronic resource]. — Mode of access: <https://store.digilentinc.com/nexys-4-artix-7-fpga-trainer-board-limited-time-see-nexys4-ddr/>. — Digilent, Inc, 2020. — Date of access: 30.10.2020.
3. Quantitative and statistical performance evaluation of arbiter physical unclonable functions on FPGAs / Y. Hori [et al.] // Proc. Int. Conf. on Reconfig. Comput. and FPGAs (ReConFig'2010). — Cancun, Mexico, 2010. — P. 298–303.

# АНАЛИЗ УСТОЙЧИВОСТИ ФИЗИЧЕСКИ НЕКЛОНИРУЕМОЙ ФУНКЦИИ ТИПА АРБИТР К КРИПТОГРАФИЧЕСКИМ АТАКАМ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ ДОЛГОЙ КРАТКОСРОЧНОЙ ПАМЯТИ

Шинкевич Н.Н., Шамына А.Ю.

Кафедра программного обеспечения информационных технологий,  
Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: nn5h@yahoo.com, shamyna@bsuir.by

*Рассматриваются способы криптографических атак на классические реализации физически неклонированных функций типа арбитра (АФНФ) с использованием ИНС выбранной конфигурации, экспериментально подтверждается эффективность выбранного подхода и необходимость использования дополнительных решений для повышения устойчивости к подобному рода атакам. Предложен способ атаки на основе сетей долгой краткосрочной памяти (NNLSTM) на АФНФ, реализованных на FPGA. Экспериментальные результаты предложенного метода NNLSTM показывают его высокую точность прогноза, до 99,96%.*

## ВВЕДЕНИЕ

В настоящее время существует проблема обеспечения цифровой безопасности, а именно: защиты устройств и их проектных описаний от несанкционированного копирования, идентификации устройств и т.д. Использование физически неклонированных функций (ФНФ) [1] позволяет решить перечисленные выше проблемы с высокими показателями случайности, уникальности и стабильности в зависимости от решаемой задачи, и в то же время с низкими аппаратными и финансовыми затратами. Широкое распространение получили ФНФ типа арбитра (АФНФ) благодаря относительной простоте реализации и сравнительно низким аппаратными затратами. Однако распространение сигналов по блоку симметричных путей (БСП) АФНФ и, следовательно, значения задержек имеют линейную природу [2], что является потенциальной уязвимостью для криптографических систем, в которых они применяются [3, 4]. В работах [5, 6] описываются способы атак, основанные на применении методов машинного обучения. Самыми эффективными при моделировании классических АФНФ являются: линейная регрессия, классификация при помощи SVM, нейронные сети. В работах [5, 6] рассматриваются атаки с использованием ИНС, которые представляют собой полностью связную структуру из плотных (Dense) слоев. Каждый такой слой обрабатывает данные без учета порядка, применяя к обучающей выборке операцию взвешенной суммы (формула 1). Такой подход является наиболее распространенным [5, 6], представленные ИНС отличаются числом плотных слоев, функциями активации, наличием либо отсутствием слоев предобработки данных, но глобально практически идентичны по своей структуре, и представляют со-

бой ничто иное, как модификации полностью связанных нейронных сетей. В настоящей статье авторами предлагается другая структура ИНС, учитывающая факт наличия временных задержек. Экспериментально подтверждается эффективность использования предложенной конфигурации ИНС.

## I. ОПИСАНИЕ КОНФИГУРАЦИИ ИНС

Выбранная авторами конфигурация ИНС состоит из слоев следующих типов:

- Embedding, преобразовывает положительные целые числа (индексы) в плотные (dense) векторы фиксированного размера;
- LSTM, особый тип РНС, способный обучаться долговременным зависимостям;
- Bidirectional, или двунаправленный слой, необходим для определения выходной последовательности с учетом всего контекста, а не только его линейной интерпретации;
- Dense, вычисляет  $f_a$  от взвешенной суммы (формула 1);
- Dropout, предотвращает переобучение сети путем удаления некоторой доли ответов случайным образом.

$$Y = f_a(X * W + B), x_i \in X, y_i \in Y, \forall i \in N \quad (1)$$

где  $W$ ,  $B$  - матрицы весов и смещений соответственно,  $f_a$  - функция активации. Чтобы воспользоваться преимуществами Embedding слоев, а именно возможностью формирования начального представления для каждого  $i$ -го элемента  $j$ -й последовательности входных данных, необходимо представить их в символьном формате. Далее с данными необходимо произвести следующие манипуляции:

1. балансировка выборки;
2. преобразование бинарной строки challenge по формулам 2, 3.

$$B_i = A, b_i == 0 \quad (2)$$

$$B_i = B, b_i = 1 \quad (3)$$

Схема построенной нейронной сети представлена на рисунке 1.

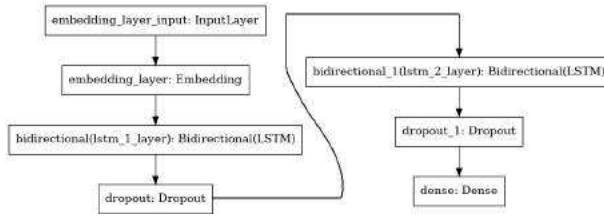


Рис. 1 – Схема модели NN LSTM

## II. ПРОВЕДЕНИЕ ЭКСПЕРИМЕНТА И СБОР ДАННЫХ

Для оценки устойчивости АФНФ различных реализаций к рассматриваемым способам атак было создано несколько наборов экспериментальных данных: собранных как с аппаратной реализации АФНФ на FPGA Artix 7, так и с параметрической модели «Post place & route», созданной с использованием САПР Xilinx 14.7 и HDL языка Verilog, а также средства моделирования ISim. Время обучения SVM и>NNLSTM было ограничено 20 эпохами для каждого тестируемого подхода.

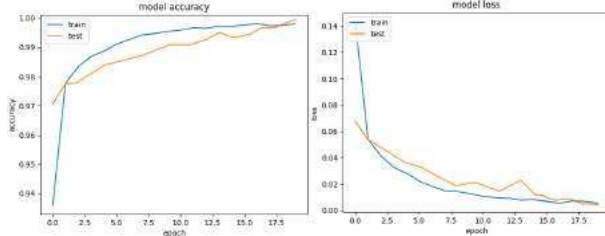


Рис. 2 – Изменение а – точности; б – ошибки модели в процессе обучения

Для достижения точности 99.96% потребовалось 20 эпох. При дообучении модели за 20 дополнительных эпох точность колеблется в пределах от 98.07 до 99.972% на тренировочной выборке, откуда следует, что незначительный прирост точности в 0.012% не оправдывает временные затраты. Процесс обучения, в частности графики изменения точности и ошибки на тренировочной и тестовой выборках представлен на рис. 2а, 2б.

Таблица 1 – Точность моделирования АФНФ на первом наборе данных

|        | SVM    | SVM    | NN LSTM | NN LSTM |
|--------|--------|--------|---------|---------|
|        | 64bit  | 128bit | 64bit   | 128bit  |
| 5000   | 54.67% | 49.93% | 52.11%  | 53.16%  |
| 10000  | 56.67% | 51.19% | 95.01%  | 98.28%  |
| 100000 | 57.77% | 55.52% | 99.15%  | 99.96%  |

Таблица 2 – Точность моделирования АФНФ на втором наборе данных

|        | SVM    | SVM    | NN LSTM | NN LSTM |
|--------|--------|--------|---------|---------|
|        | 64bit  | 128bit | 64bit   | 128bit  |
| 5000   | 52.92% | 31.89% | 61.81%  | 45.93%  |
| 10000  | 53.99% | 33.16% | 97.89%  | 97.20%  |
| 100000 | 58.11% | 52.08% | 99.94%  | 99.93%  |

## III. ЗАКЛЮЧЕНИЕ

Исходя из результатов (табл. 1, 2), можно сделать вывод:>NNLSTM успешно смоделировала несколько видов АФНФ, более того, сеть смогла найти зависимости в данных, используя выборку из  $10^4$  примеров, а при наличии  $10^5$  примеров сеть достигла почти 100%-й точности, чего нельзя сказать о методе SVM [7]; как показала практика, ему недостаточно 20 эпох обучения для достижения приемлемого уровня точности (предел был достигнут за 94 эпохи, точность 97.16% для набора данных из  $10^5$  примеров). Более того, при моделировании более сложной АФНФ точность метода SVM падала, тогда как на>NNLSTM это почти не повлияло (наблюдается лишь малое снижение точности). Стоит отметить, что все выборки являются относительно небольшими по размеру; обычно ИНС требуется  $10^6$  примеров. Полученные результаты демонстрируют необходимость применения дополнительных решений при использовании АФНФ в качестве криптографического примитива. Например, в работе [8] предлагается использование нелинейной обфускации запросов к АФНФ при помощи MISR.

## IV. СПИСОК ЛИТЕРАТУРЫ

1. Security with Noisy Data / P. Tuyls, B. Skoric, T. Kevenaar. – Switzerland : Springer, 2007. – 344 p.
2. Physical unclonable functions / Y. Gao, S. F. Al-Sarawi, and D. Abbott, Nature Electron., vol. 3, no. 2, pp. 81–91, Feb. 2020.
3. Modeling attacks on physical unclonable functions / U. Ruhrmair [et al.] // Proc. 17th ACM conf. on Comp. and comm. secur. (CCS'10). – Chicago, USA, 2010. – P. 237–249.
4. PUF modeling attacks on simulated and silicon data / U. Ruhrmair [et al.] // IEEE Trans. on Inform. Forens. and Secur. – 2013. – № 8 (11). – P. 1876–1891.
5. Going Deep: Using deep learning techniques with simplified mathematical models against XOR BR and TBR PUFs / M. Khalafalla, M. A. Elmohr and C. Gebotys, 2020 IEEE International Symposium on Hardware Oriented Security and Trust (HOST), 2020, pp. 80–90.
6. Deep Learning based Model Building Attacks on Arbiter PUF Compositions / Santikellur, P., Bhattacharyay, A., & Chakraborty, R. (2019). IACR Cryptol. ePrint Arch., 2019, 566.
7. Security evaluation and enhancement of bistable ring PUFs / X. Xu [et al.] // Proc. Int. Worksh. on Rad. Freq. Ident.: Secur. and Priv. Iss.(RFIDSec'2015). – New York, USA, 2015. – P. 3–16.
8. S. S. Zalivaka, A. A. Ivaniuk and C. Chang, "Low-cost fortification of arbiter PUF against modeling attack"2017 IEEE International Symposium on Circuits and Systems (ISCAS), 2017, pp. 1-4.



Секция «Математическое  
моделирование и компьютерная  
графика»

Section «Mathematical modeling and  
computer graphics»

# АЛГОРИТМ ВЫЯВЛЕНИЯ НЕИСПРАВНОСТЕЙ МЕХАНИЗМА, ОСНОВАННЫЙ НА ПРЕОБРАЗОВАНИИ ВИБРАЦИОННОГО СИГНАЛА

Бондарева Т. О., Лапицкая Н. В.

Кафедра программного обеспечения информационных технологий,  
Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: tatyana.bondareva.2016@mail.ru, lapan@bsuir.by

*Исследуются методы выявления неисправностей в механизма. Приводится описание таких методов как спектральный анализ Фурье, Вейвлет-анализ и поднимается проблематика их ограничений. Исследуется новый метод выявления неисправностей в механизма, основанный на Гильбертовом преобразовании. Эффективность данного метода иллюстрируется путем сравнительной оценки результатов работы Вейвлет преобразования и Гильбертова преобразования.*

## ВВЕДЕНИЕ

Анализ вибрационных процессов, возникающих в промышленном оборудовании, позволяют достаточно полно отобразить техническое состояние многих деталей и характер протекания рабочего процесса. Поэтому вибрационная диагностика не только позволяет обнаруживать неисправности, но и выявлять их причины, например повышенные динамические нагрузки или переменные напряжения. Известно, что определение необходимых информативных параметров и характеристик вибрации является весьма сложной задачей, так как это связано с выбором соответствующих методов обработки механических колебаний, учитывающих их реальную структуру и свойства. Спектральный анализ – это метод обработки сигналов, который позволяет выявить частотный состав сигнала. Анализ спектрограмм виброускорения позволяет распознать повреждения на ранней стадии.

## ПРОБЛЕМА ВЫБОРА АЛГОРИТМА СПЕКТРАЛЬНОГО АНАЛИЗА

Широко известными алгоритмами анализа вибрационных сигналов В настоящее время наиболее распространенными алгоритмами для обработки вибрационных сигналов различного происхождения являются методы спектрального анализа, основанные на дискретном преобразовании Фурье, или Вейвлет преобразование.

Преобразование Фурье дает мощный математический аппарат, позволяющий эффективно решить ряд практически важных задач. И, что наиболее важно, преобразование Фурье относится к категории физически обоснованных математических моделей, так как оно физически реализуется различными устройствами. Но существует ограничение данного преобразование: исследуемые данные должны быть стационарными.

Основная идея вейвлет преобразования заключается в том, чтобы для поиска локальной особенности использовать разложение по функциям, похожим на искомую особенность. Такой

подход предполагает представление функции одной координаты в виде функции двух координат – пространственной и масштабной. Однако применение вейвлет-преобразования требует предварительного выбора типа вейвлета и центральной частоты его частотной характеристики.

С учетом недостатков рассмотренных выше алгоритмов было принято решение проведении исследований по применению модовой декомпозиции для анализа нестационарных сигналов.

## ГИЛЬБЕРТОВО ПРЕОБРАЗОВАНИЕ

Классический метод модовой декомпозиции основан на предположении, что любые данные состоят из различных режимов колебаний. Каждый режим, линейный или нелинейный, стационарный или нестационарный, представляет простое колебание, которое в определенной степени «симметрично» относительно локального среднего значения, а, следовательно, имеет экстремумы и нулевые пересечения.

Декомпозиция на модовые функции представляет собой итерационную процедуру.

Шаг 1. Определение всех экстремумов (максимумов и минимумов) исходного сигнала  $Y(t)$ .

Шаг 2. Построение верхней  $Y_{max}(t)$  и нижней огибающей  $Y_{min}(t)$  по полученным минимальным и максимальным значениям строится.

Шаг 3. Вычисление среднего значение огибающих по формуле:

$$m_1(t) = \left( \frac{Y_{min}(t) + Y_{max}(t)}{2} \right).$$

Шаг 4. Определение первого приближения  $h_{11}$  к первой функции эмпирической моды сигнала  $Y(t)$ :

$$h_{11}(t) = Y(t) + m_1(t).$$

Принимая вместо функции  $Y(t)$  функцию  $h_{11}(t)$  и повторяя шаги с первого по четвертый, находим второе приближение к первой функции моды  $h_{12}(t)$ . Аналогичным образом находим и



следующие приближения  $h_{1k}(t)$  функции моды  $C_1(t)$ . Итерации продолжаются до тех пор, пока нормализованная квадратичная разность между двумя приближениями  $h_{1k}(t)$  и  $h_{1k-1}(t)$ , не станет меньше некоторого предельного значения.

Последнее значение  $h_{ik}(t)$  итераций принимается за первую, наиболее высокочастотную моду  $C_1(t) = h_{ik}(t)$  семейства модовых функций, которая непосредственно входит в состав исходного сигнала  $Y(t)$ . Это позволяет вычистить  $C_1(t)$  из состава сигнала и оставить в нем более низкочастотные составляющие:

$$Y(t) - C_1(t) = r_1(t).$$

Для получения второй эмпирической моды  $C_2(t)$  над остатком  $r_1(t)$  повторяются те же преобразования, что и для получения первой моды  $C_1(k)$ .

В результате разложения сигнала  $Y(t)$  на модовые функции, он может быть представлен в виде:

$$Y(t) = \sum_{i=1}^n C_i(t) + r_n.$$

#### АНАЛИЗ РЕЗУЛЬТАТОВ ГИЛЬБЕРТОВОГО ПРЕОБРАЗОВАНИЯ

Для анализа работы Гильбертового преобразования проведем сравнение результатов его работы с результатами работы вейвлет-преобразования. С электродвигателя лабораторной установки были сняты показатели виброускорения. Полученный сигнал содержал кратковременные возмущения на некоторых участках. Рассмотрим один из таких участков в качестве исследуемого сигнала.

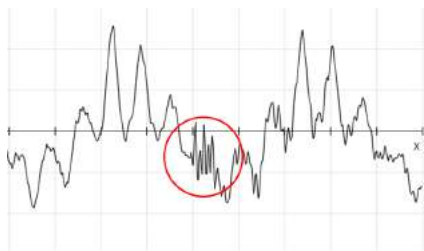


Рис. 1 – Вибросигнал с кратковременным возмущением

Произведем выполнение Вейвлет преобразования исследуемого сигнала. Путем лабораторных испытаний были найдены метод и частота, для которых возможно выявление возмущений.

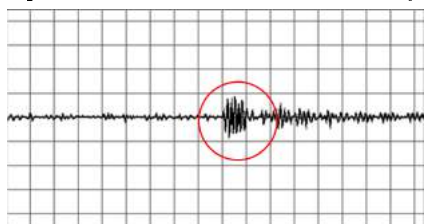


Рис. 2 – Вейвлет преобразование вибросигнала методом Gaus4p с частотой 2100

Выполним Гильбертово преобразование для исследуемого сигнала. Его результаты показали, что выявление возмущений можно получить уже при анализе первой модовой декомпозиции.

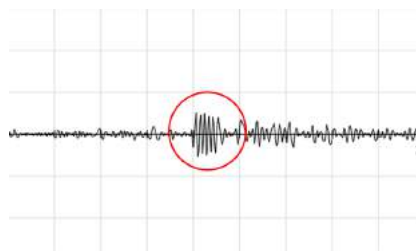


Рис. 3 – Первая мода вибросигнала

Основываясь на данных результатах можно сказать модовая декомпозиция и вейвлет-преобразования одинаково хорошо справляются с задачей нахождения возмущений.

#### ЗАКЛЮЧЕНИЕ

Из анализа преимуществ и недостатков вейвлет-преобразования, кратковременного Фурье-спектрометра и Гильберт-Хуанг преобразования можно сделать представленные ниже выводы.

Преобразование Фурье ограничено по сравнению с преобразованием Вейвлета и преобразованием Гильберта-Хуанга при анализе нелинейных и неустойчивых сигналов. Вейвлет-преобразование было лучшим доступным анализом нелинейных данных перед введением Гильберта-Хуанга преобразования. Преобразование Гильберта-Хуанга имеет лучшие временные и частотные разрешающие способности, чем вейвлет-преобразование. Преобразование Гильберта-Хуанга оказывается новым революционным методом анализа нелинейных и неустойчивых данных сигнала.

1. Алексеев, С. П. Борьба с шумом и вибрацией в машиностроении. / С. П. Алексеев, А. М. Казаков, Н. Н. Колотылов // М.: Машиностроение. 1970. – 208 с.
2. Кулаичев, А. П. Критика вейвлет анализа ээг / М. М. Медведев, Н. Н. Бурова // Актуальные проблемы гуманитарных и естественных наук. 12-1. – 2016. – С. 47–57.
3. Кан, Ш. Ч. Анализ нестационарных сигналов на основе преобразования Гильберта-Хуанга / Ш. Ч. Кан, А. В. Микулович, В. И. Микулович // Информатика. – 2010. – № 2. – С. 36–47.
4. The empirical mode decomposition method and the Hilbert spectrum for non-stationary timeseries analysis / N.E. Huang [et al.] // Proc. R. Soc. Lond. – 1998. – A454. – P. 903–995

# РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ АВТОМАТИЗИРОВАННОГО РАБОЧЕГО МЕСТА ВОЕННОСЛУЖАЩЕГО КУРСОВОГО ЗВЕНА

Герасимов В. А.

Кафедра вычислительных методов и программирования,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: v.gerasimov@bsuir.by

*В данной статье рассматривается разработка программного обеспечения автоматизированного рабочего места военнослужащего курсового звена военного факультета в УО БГУИР. Проанализирована предметная область применения программного обеспечения, выделены функциональные требования. Описаны этапы проектирования базы данных, выбор средств для разработки программного обеспечения.*

## ВВЕДЕНИЕ

Повседневная задача военного образования заключается в обеспечении курсантов соответствующими знаниями, навыками и умениями, поиске эффективных путей мотивации учебной и самостоятельной деятельности.

Офицер, работающий с военнослужащими, постоянно получает разнообразную информацию из различных источников информации. Чтобы держать ее в поле своего зрения, он должен делать записи, анализировать, классифицировать собранный материал. Это позволяет командиру выявить тенденции, закономерности становления и развития личности, увидеть социально-психологические явления, происходящие в коллективе, сделать правильные выводы, дать рекомендации командному составу подразделения по индивидуальной и воспитательной работе с подчиненными.

Изучение воинского коллектива предполагает оценку следующих показателей:

- основные сведения коллективе (социально-демографические данные, образовательный уровень, возраст, национальность, род деятельности до поступления, призыва в ВС, условия воспитания и т.д.);
- дисциплинированность каждого военнослужащего и тенденции, характеризующие состояние воинской дисциплины в коллективе;
- успеваемость в учебе.

На основании полученной информации офицер ведет следующую документацию:

- личное дела курсанта;
- служебную карточку курсанта с взысканиями и поощрениями;
- график нарядов взводов;
- график увольнений курсантов.

## I. ОПИСАНИЕ ФУНКЦИОНАЛЬНЫХ ТРЕБОВАНИЙ

Задачей разрабатываемого программного обеспечения является систематизация данных о курсанте (социально-демографические данные,

уровень образования, возраст, национальность, род деятельности до поступления, призыва в Вооруженные Силы, условия воспитания), информации о поощрениях и взысканиях, наложенных на курсанта в период обучения, информации о нарядах. Для анализа данных необходимо представить инструмент, который позволяет выявлять социально-демографических особенностей в коллективе.

Описание ролей в рамках разрабатываемой системы, а также краткий перечень выполняемых ими функций представлено в таблице 1.

Таблица 1 – Классы и характеристики пользователей

| Роль пользователя     | Класс пользователя          | Описание класса                                                                                                                                              |
|-----------------------|-----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Курсовой офицер       | Авторизованный пользователь | Осуществляет ввод информации о курсантах, успеваемости, дисциплине, нарядах и увольнениях                                                                    |
| Командир батальона    | Авторизованный пользователь | Производит анализ информации. Имеет возможность корректировать введенную информацию, просматривать графики успеваемости, социально-демографическое положение |
| Администратор системы | Авторизованный пользователь | Осуществляет управление работой системы. Имеет возможность внесения изменения в настройки системы, регистрации новых пользователей                           |

Проект должен включать в себя следующие функции:

- добавление информации о курсах, группах и курсантах с возможностью проверки корректности вводимых данных, редактирования и удаления введенной информации;
- добавление дополнительной информации о курсантах в виде информации об успеваемости, поощрениях, взысканиях, нарядах, увольнениях с возможностью проверки корректности вводимых данных, редактирования и удаления введенной информации;
- сохранение добавленной информации в базу данных;
- сохранение информации из базы данных в текстовый файл;
- загрузка информации из файла в базу данных с возможностью проверки корректности вводимых данных из файла;
- формирование и управление базой данных с личной информацией о курсантах;
- вывод информации из базы данных на основании введенной информации;
- формирование графиков по динамике изменения успеваемости, количества поощрений и взысканий, нарядов за определенный период;
- авторизация и аутентификация пользователей;
- возможность добавления (редактирования)/удаления пользователей после авторизации;
- возможность ограничения функций приложения согласно роли пользователя.

## II. ОСНОВНЫЕ КОМПОНЕНТЫ СИСТЕМЫ

В разрабатываемой системе сервер базы данных представлен Microsoft SQL Server LocalDB, который обеспечивает высокую производительность выполнения запросов, масштабируемость, безопасность и интегрируемость.

В качестве клиента выступает классическое приложение, построенное по технологии Windows Presentation Foundation.

Обращение приложения к базе данных осуществляется посредством фреймворка Entity Framework со способом взаимодействия с базой Code-First[1]. Этот подход предоставляет ряд существенных преимуществ: вам не нужно беспокоиться о коде доступа к данным, вам не нужно знать деталей работы СУБД SQL Server и синтаксиса языка запросов T-SQL, вместо этого вы работаете с таблицами базы данных как с классами, с полями этих таблиц - как со свойствами классов, а синтаксис SQL-запросов, который в ADO.NET раньше нужно было вставлять в код в виде команд, заменен на более удобный подход с LINQ. Entity Framework берет на себя обязанности по преобразованию кода в SQL-инструкции.

## III. РАЗРАБОТКА СТРУКТУРЫ БАЗЫ ДАННЫХ

Проектируемая база данных должна содержать личные данные курсанта, данные об успеваемости, его поощрениях и взысканиях, нарядах. Для авторизации и аутентификации пользователей необходимо хранить данные пользователей в базе данных. Диаграмма сущностей изображена на рисунке 1.

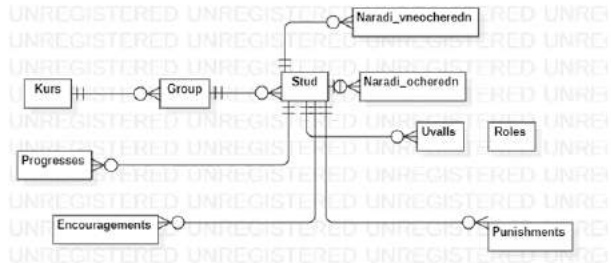


Рис. 1 – Диаграмма сущностей

Подход Code-First основан на следующем принципе: разработчик создает класс модели данных, которые будут храниться в базе данных, а затем Entity Framework по этой модели генерирует базу данных и ее таблицы. На рисунке 2 представлена база данных, сгенерированная после запуска приложения.

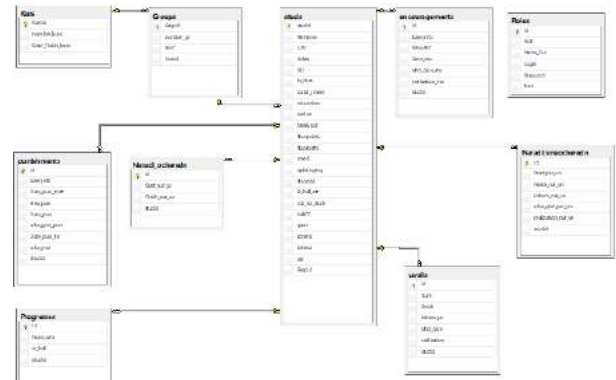


Рис. 2 – Схема сгенерированной базы данных

## ЗАКЛЮЧЕНИЕ

Разработанное программное обеспечение позволяет значительно упростить процесс ведения учета личных данных курсантов, данных успеваемости, дисциплинарной практики, количеству увольнений и нарядов, создавать отчеты по социально-демографическому положению батальона по имеющимся сведениям, обеспечить доступ к данным для пользователей разного уровня.

## СПИСОК ЛИТЕРАТУРЫ

1. Работа с Entity Framework 6 [Электронный ресурс] / – Режим доступа: <https://professorweb.ru/my/entity-framework/6/level1/> – Дата доступа: 19.10.2021.

# МОДИФИЦИРОВАННЫЙ АЛГОРИТМ СЖАТИЯ ДАННЫХ

Гуринович А. Б., Митьковец Л. В., Сидоров Д.  
Кафедра информационных технологий и управления,

Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь

E-mail: gurinovich@bsuir.by, lidiatommo@gmail.com, sam65th@mail.ru

Сжатие данных применяется для более рационального использования устройств хранения и передачи данных путем устранения избыточности. Оно будет оставаться в сфере внимания из-за все более возрастающих объемов хранимых и передаваемых в ЭВМ данных, кроме того, его можно использовать для преодоления некоторых физических ограничений, таких как, например, сравнительно малая пропускная способность линии.

## ВВЕДЕНИЕ

Исследуемый модифицированный алгоритм компрессии является алгоритмом сжатия данных без потерь. Предлагаемый алгоритм J-битного кодирования (ЖВЕ) [4] работает путем манипулирования битами данных для уменьшения размера и оптимизации входных данных для другого алгоритма. Идея этого алгоритма состоит в том, чтобы разделить входные данные на две группы: первая будет содержать исходный ненулевой байт, а вторая — битовое значение, объясняющее положение ненулевого «1» и нулевого «0» байтов. Затем обе группы могут быть сжаты отдельно с помощью другого алгоритма сжатия данных для достижения максимальной степени сжатия.

### I. ПОШАГОВЫЙ ПРОЦЕСС J-БИТНОГО КОДИРОВАНИЯ.

Пошаговый процесс сжатия можно описать следующим образом:

- Считывание входных данных по байтам.
- Определение прочитанных байт как «0» или «1» байт.
- Запись «0» байт в данные I и запись бита «1» во временные байтовые данные, или бита «0» во временные байтовые данные для нулевого входного байта.
- Повторение первых трех шагов до тех пор, пока временные байтовые данные не будут заполнены 8 битами данных.
- Если временные байтовые данные заполнены 8 битами, то запись байтового значения производится в данные II.
- Очистка временных байтовых данных
- Повторение первых шести шагов до тех пор, пока не будет достигнут конец файла.
- Запись комбинированных выходных данных:
  1. Запись исходной входной длины.
  2. Запись данных I.
  3. Запись данных II.

На рис. 1 показан визуальный пошаговый процесс сжатия для J-битного кодирования. Вставленная исходная длина входного сигнала в начало выходного сигнала будет использоваться в качестве информации для размера данных I и данных II.

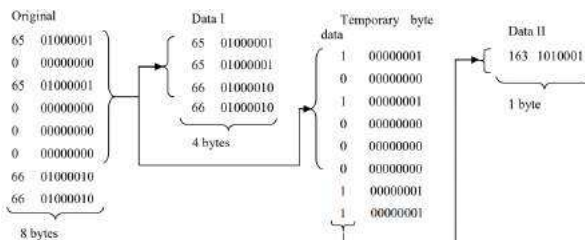


Рис. 1 – Алгоритм ЖВЕ

Пошаговый процесс декомпрессии можно описать следующим образом:

- Считывание исходной длины входного сигнала.
- Если было сжато отдельно, то распаковка данных I и данных II (необязательно).
- Побитное считывание данных II.
- Определение, является ли бит чтения «0» или «1».
- Запись на выход: если бит чтения равен «1» то чтение и запись данных I на выход, если бит чтения равен «0» то запись нулевого байта на выход.
- Повторение шагов 2-5 до тех пор, пока не будет достигнута исходная длина входного сигнала.

### II. СРАВНЕНИЕ РЕЗУЛЬТАТОВ В КОМБИНАЦИИ С ДРУГИМИ АЛГОРИТМАМИ СЖАТИЯ

В сравнении используются методы компрессии: кодирования длин серий (RLE) [3], преобразования Берроуза-Уилера (BWT) [2], уменьшения избыточности (MTF) [3] и арифметического кодирования (ARI) [3].

Различные комбинации алгоритмов сжатия данных тестируются с пятью типами файлов. Каждый тип состоит из 80 образцов. Каждый образец имеет разный размер, чтобы показать ре-

альное состояние файловой системы. Все образцы не сжаты; они включают в себя необработанные растровые изображения и звук без сжатия с потерями. Образцы для эксперимента приведены на рис. 2.

| Ном. | Тип                   | Кол-во | Хар-ки     |
|------|-----------------------|--------|------------|
| 1    | Растровое изображение | 80     | Raw 8 bit  |
| 2    | Растровое изображение | 80     | Raw 24 bit |
| 3    | Текстовый документ    | 80     |            |
| 4    | Исполняемый файл      | 80     |            |
| 5    | Звук WAVE             | 80     | WAV        |
| 6    | Видео AVI             | 80     | VBR        |

Рис. 2 – Используемые в сравнении образцы

Последовательность комбинаций методов компрессии:

1. BWT+MTF+RLE;
2. BWT+RLE+ARI;
3. RLE+BWT+MTF+RLE+ARI;
4. RLE+BWT+MTF+JBE+ARI.

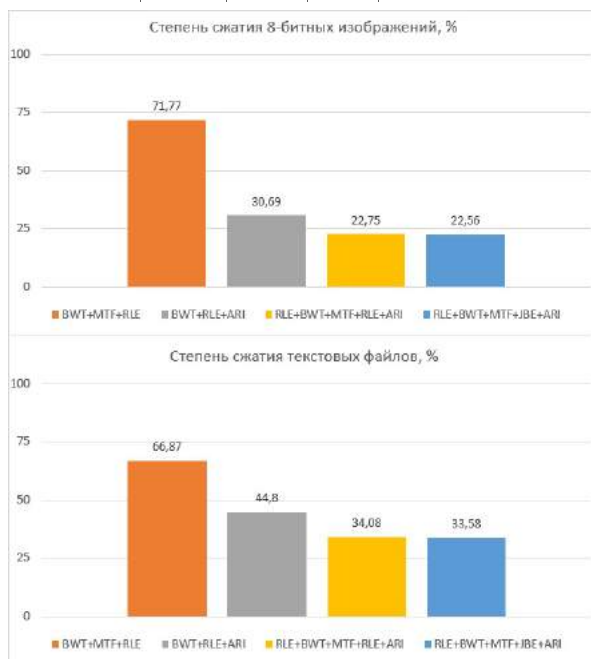


Рис. 3 – Сравнение результатов сжатия

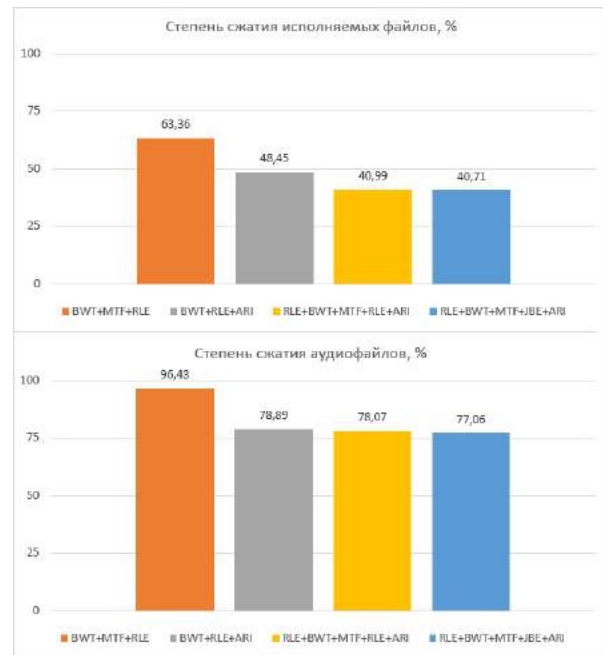


Рис. 4 – Сравнение результатов сжатия (продолжение)

На рис. 3 и рис. 4 показано, что используемые образцы различных типов данных сжимаются с большей степенью компрессии в сочетании с предложенным алгоритмом.

### III. ЗАКЛЮЧЕНИЕ

В исследовании проведен эксперимент с использованием 6 типов файлов с 80 различными размерами для каждого типа. В результате протестировано и проведено сравнение 4 комбинаций алгоритмов. Предложенный алгоритм дает лучшую степень сжатия при вставке между преобразованиями «move-to-front transform» (MTF) и арифметическим кодированием (ARI). Рассматриваемый алгоритм имеет перспективу внедрения в структуру других алгоритмов сжатия данных.

### IV. СПИСОК ЛИТЕРАТУРЫ

1. Capo-chichi, E. P., Guyennet, H. and Friedt, J. K-RLE a New Data Compression Algorithm for Wireless Sensor Network. In Proceedings of the 2009 Third International Conference on Sensor Technologies and Applications.
2. Nelson, M. 1996. Data compression with Burrows-Wheeler Transform. Dr. Dobb's Journal.
3. Springer, Handbook of Data Compression Fifth Edition.
4. Agus Dwi Suarjaya. Information Technology Department Udayana University. Bali, Indonesia. A New Algorithm for Data Compression Optimization.

# МЕТОДЫ ВЫДЕЛЕНИЯ ИНФОРМАЦИОННЫХ ПРИЗНАКОВ РЕЧЕВЫХ СИГНАЛОВ

Езовит А. В., Гуринович А. Б.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: ezovit99@yandex.ru, gurinovich@bsuir.by

*В настоящее время важной задачей в области речевой обработки является выделение информационных признаков сигнала. Достижения в данной сфере могут применяться для идентификации личности, её описания, сбора информации о физическом состоянии и роде деятельности. В данной статье приведены методы, которые могут использоваться для решения задачи выделения информационных признаков.*

## ВВЕДЕНИЕ

Важной задачей в области речевой обработки является выделение информационных признаков сигнала. Информационными признаками являются характеристики голоса и речи, которые уникальны для каждого человека. Их выделение из сигнала позволит составить «портрет» речи человека, позволяющий определить кем он является и при необходимости идентифицировать его. Примерами информационных признаков в голосовом сигнале (обработка голоса) являются основной тон (характеризующий гендер человека) и обертоны (характеризующий спикера)[1]. В речевом сигнале — темп и благозвучность речи, которые, в отличие от приведенных выше голосовых признаков, не обоснованы (или обоснованы не полностью) анатомией каждого конкретного человека и являются следствием привычек и тренировок[2].

Задачей, схожей с выделением признаков, является задача идентификации человека. При решении этой задачи собирается вектор признаков, являющийся «слепком голоса» и представляющий из себя смесь из разных признаков. Таким образом, цель задачи по выделению — разделение этой смеси на составляющие ее информационные признаки.

Задачи по выделению признаков актуальны, например, для подтверждения личности клиентов банков, установления личности в криминалистике и других областях.

## I. ХАРАКТЕРИСТИКИ ГОЛОСА

Первоначально для выполнения задачи выделения информационных признаков в речевом сигнале определяют, что может служить определяющей характеристикой голоса. Признаки, заключенные в речевом сигнале, делятся на две группы: низкоуровневые/голосовые (признаки, характеризующие голос и обусловленные анатомией человека) и высокоуровневые/речевые (признаки, характеризующие речь, связанные с манерой общения)[3].

К низкоуровневым относятся:

- Сила голоса (амплитуда)

- Высота голоса (частота)
- Тембр голоса (окраска голоса, задаваемая обертонами)
- Полетность (слышимость голоса на большом удалении от источника)
- Благозвучность голоса (звонкость и частота звучания без призвуков)

К высокоуровневым относятся:

- Подвижность (способность человека к изменению параметров голоса)
- Тон голоса (эмоциональная окраска)
- Темп речи (скорость, с которой говорит человек)
- Артикуляция и дикция (отвечают за четкость произнесения звуков в соответствии с нормами языка)

## II. ПРЕДВАРИТЕЛЬНАЯ ОБРАБОТКА ИСХОДНОГО СИГНАЛА

Для выделения информационных признаков в речевом сигнале проводится анализ отцифрованного сигнала. Анализ может проводиться непосредственно над амплитудно-временным представлением сигнала или над его спектральным видом. Процесс обработки и приведения к спектральной форме производится в несколько этапов:

1. Фильтрация сигнала от выбросов, вызванных микрофоном (данные выбросы зависят от качества микрофона и могут поднимать средний уровень сигнала на некоторое значение, что в свою очередь приводит к проблемам обработки) и низкочастотных шумов (данные шумы особо выделяются в местах спада энергии в амплитудно-временной области из-за чего могут быть восприняты за голосовую активность и вызвать проблемы при анализе).
2. Нормализация сигнала (процесс выравнивания громкости сигнала относительно какого-либо эталона, может применяться пиковая и средняя нормализация, определяющая будет с эталоном сравниваться пи-

ковое или среднее значение соответствующе).

3. Разбиение сигнала на временные кадры (длина кадра выбирается в зависимости от длины интервала времени, на котором происходит всплеск мощности, и обычно находится в диапазоне 10-150 мс).
4. Применение весовой функции (например окно Хэмминга) для уменьшения искажений (при этом выделяется центральная часть кадра и происходит плавное затухание по краям, что упрощает дальнейшую обработку).
5. Осуществления преобразования Фурье.

После окончания преобразования осуществляется переход в логарифмический масштаб, к которому чувствительно человеческое ухо (пример Мэл-шкала, данная шкала имеет линейную природу на диапазоне до 1000 Гц и логарифмическую для частот выше).

Первичная обработка сигнала является важным шагом на пути к последующему анализу, поскольку правильно проведенная обработка может существенно снизить мощности необходимые на анализ.

### III. МЕТОДЫ ВЫДЕЛЕНИЯ ПРИЗНАКОВ

При решении задачи распознавания спикера, по полученному спектру собирается суммарный вектор признаков, для которого в последующем на основе метрики (определяется расстояние между векторами, например евклидово расстояние, L1-норма, расстояние Махалонобиса) или оценки вероятности (модель гауссовских смесей) определяется схожесть с имеющимся вектором (характеризующим голос спикера) и делается вывод, являются ли говорящие одним человеком[4].

Для эффективного решения задачи выделения признаков предлагается следующее: аналогичным образом собирается вектор признаков, однако в последующем данный набор не сравнивается с уже имеющимся вектором, а аналогичными методом определения метрического расстояния или оценки вероятности проверяется входимость признака в ранее определенную группу. Например, при предварительной подго-

товке выборка из мужских, женских и детских голосов будет обработана и проанализирована, после чего для каждого голоса будет выделен свой вектор признаков, на основе которых произведется сравнение, определяющее группы (интервал), входимость в которую означает, наличие определенного признака в образце голоса (в данном примере мужской, женский, детский). Подобным подходом может оцениваться эмоциональность в голосе, сила, чистота звучания и другие признаки.

Эффективно будет добавить еще один выделяемый признак – скорость произношения, которая может отличаться от человека к человеку. Оценку данного признака можно произвести на основе спада амплитуд во временной или частотной области.

### IV. ЗАКЛЮЧЕНИЕ

Предложенные методы позволят провести анализ голоса спикера на входимость его в различные группы, что в свою очередь позволяет выделять определенные информационные признаки. Очевидно, что при составлении групп, сигнализирующих о наличии того или иного признака, важным моментом, влияющим на правильность определения, является выборка заранее обработанных голосов, на основе которых строится интервал группы. Результаты выделения в совокупности с моделями, построенными в других областях знаний, позволят не только описать голос человека, но и установить другие закономерности. Например, физическое и эмоциональное состояние или возможный род деятельности. Это позволит эффективно решать проблему выделения информационных признаков речевого сигнала.

### V. СПИСОК ЛИТЕРАТУРЫ

1. Григорьев Е.И., Тычинина В.М. Звуки речи и их коммуникативная функция : учебное пособие для студентов филологических специальностей, аспирантов и преподавателей / Е.И. Григорьев, В.М. Тычинина. — Тамбов, ТГУ им. Г.Р.Державина. — 84 с.
2. 4brain [Электронный ресурс]. – Москва, 2021 – Режим доступа: <https://4brain.ru/voice/properties.php> – Дата доступа: 02.11.2021.
3. Е.А. Первушин обзор основных методов распознавания дикторов // Математические структуры и моделирование 2011, вып 24: Омск 2011. – С. 41-54.
4. habr [Электронный ресурс]. – Москва, 2021 – Режим доступа: <https://habr.com/ru/post/336516/> – Дата доступа: 02.11.2021.

# ПАРАМЕТРИЧЕСКАЯ СХЕМА МОДЕЛИРОВАНИЯ ХАРАКТЕРИСТИК ВИБРОИЗОЛЯТОРОВ ДЛЯ СИНТЕЗАТОРОВ ЧАСТОТ

Журавлёв В.И., Наумович Н.М., Бричкалевич П.И.

Центр 1.6, Белорусский государственный университет информатики и радиоэлектроники  
г.Минск, Республика Беларусь

E-mail: vadh@bsuir.by, naumovich@bsuir.by

*Приведён способ построения параметрической схемы моделирования в программном комплексе инженерного анализа для расчёта и оптимизации параметров системы виброизоляции синтезаторов частот СВЧ-диапазона.*

## ВВЕДЕНИЕ

Работа синтезаторов частоты СВЧ-диапазона часто сопряжена с воздействием внешних вибрационных воздействий, что ведёт к нарушению стабильности генерации сетки частот и появлению фазового шума [1]. В качестве элементов пассивной вибрационной защиты синтезаторов часто используются различные виды виброизолирующих креплений со смещением основной резонансной частоты [2].

Наилучшие характеристики конструкций виброизоляторов можно определить, используя программные средства, основанные на алгоритмах метода конечных элементов, конечных разностей или многотельной динамики [3]. Целесообразно использовать параметрическую схему моделирования, представляющую адаптивное перестроение расчётной модели для нахождения оптимальных выходных параметров.

## 1. СТРУКТУРА ПАРАМЕТРИЧЕСКОЙ СХЕМЫ

При использовании статического моделирования производится анализ жёсткостей моделей виброгасителей или виброизоляторов в различных направлениях, зависящих от упругих или вязкоупругих свойств материалов, а также сопоставление полученных результатов расчёта со значениями, определёнными производителем в соответствующих каталогах. Модальный тип исследований предназначен для определения соб-

ственных форм и частот колебаний, которые в свою очередь являются резонансными для каждого виброгасителя.

Воздействие случайной вибрации удобно задавать стандартным параметром спектральной плотностью мощности (PSD) [4]. Рассматривались несколько видов PSD для первоначальной настройки модели и определения корреляции параметров.

Используемая параметрическая схема с обратной связью для анализа конструкции синтезатора с использованием библиотек материалов, компонентов и средств уменьшения вибрации приведена на рисунке 1.

Редактор геометрии предназначен как для создания расчётной модели, так и для импорта и редактирования твердотельной модели, которая в дальнейшем используется при проведении расчётов.

Блок решения статической задачи (формирование преднагружения) необходим для приведения системы в положение равновесия и определения нагрузки на упругие элементы конструкции, которые будут использоваться для анализа собственных частот.

Блок модального анализа позволяет получить заданное количество форм и значений собственных частот, на основе которых решается последующая задача определения виброхарактеристик.

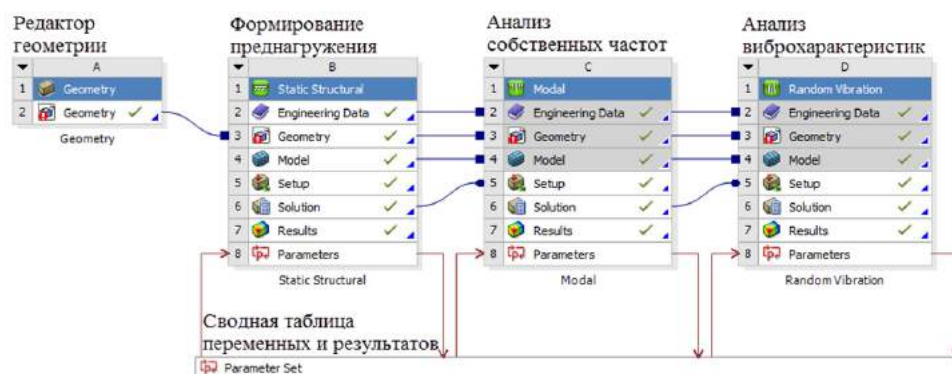


Рис. 1 – Параметрическая схема решения задачи отклика конструкции на случайную вибрацию



|    | A                      | B                                                                | C              | D                  |
|----|------------------------|------------------------------------------------------------------|----------------|--------------------|
| 1  | ID                     | Parameter Name                                                   | Value          | Unit               |
| 2  | Input Parameters       |                                                                  |                |                    |
| 3  | Static Structural (B1) |                                                                  |                |                    |
| 4  | P1                     | Longitudinal - kogu_1-1 To generator-1<br>Longitudinal Stiffness | 60             | N mm <sup>-1</sup> |
| 5  | P2                     | Longitudinal - kogu_1-1 To generator-1<br>Longitudinal Stiffness | 60             | N mm <sup>-1</sup> |
| 6  | P3                     | Longitudinal - kogu_1-1 To generator-1<br>Longitudinal Stiffness | 60             | N mm <sup>-1</sup> |
| 7  | P4                     | Longitudinal - kogu_1-1 To generator-1<br>Longitudinal Stiffness | 60             | N mm <sup>-1</sup> |
| 8  | P5                     | Longitudinal - kogu_1-1 To generator-1<br>Longitudinal Stiffness | 60             | N mm <sup>-1</sup> |
| 9  | P6                     | Longitudinal - kogu_1-1 To generator-1<br>Longitudinal Stiffness | 60             | N mm <sup>-1</sup> |
| 10 | P7                     | Longitudinal - kogu_1-1 To generator-1<br>Longitudinal Stiffness | 60             | N mm <sup>-1</sup> |
| 11 | P8                     | Longitudinal - kogu_1-1 To generator-1<br>Longitudinal Stiffness | 60             | N mm <sup>-1</sup> |
| *  | New input parameter    | New name                                                         | New expression |                    |
| 13 | Output Parameters      |                                                                  |                |                    |
| 14 | Static Structural (B1) |                                                                  |                |                    |
| 17 | Modal (C1)             |                                                                  |                |                    |
| 38 | Random Vibration (D1)  |                                                                  |                |                    |
| 39 | P39                    | Directional Acceleration Maximum                                 | 560,74         | m s <sup>-2</sup>  |
| 40 | P40                    | Directional Acceleration 2 Maximum                               | 28225          |                    |
| 41 | P41                    | Directional Acceleration 3 Maximum                               | 169,02         |                    |
| 42 | P42                    | Directional Velocity Maximum                                     | 1,238E+06      | mm s <sup>-1</sup> |
| 43 | P43                    | Directional Velocity 2 Maximum                                   | 1,832E+06      | mm s <sup>-1</sup> |
| 44 | P44                    | Directional Velocity 3 Maximum                                   | 1,238E+06      | mm s <sup>-1</sup> |
| *  | New output parameter   |                                                                  | New expression |                    |
| 46 | Charts                 |                                                                  |                |                    |

Рис. 2 – Блок управления параметрами

В блоке решения случайных вибраций определяются значения виброперемещений, виброскоростей, виброускорений при заданных характеристиках возмущающего воздействия (сигнала). Возмущающее воздействие должно быть задано зависимостью параметра PSD от частот колебаний.

Блок управления входными и выходными параметрами представляет собой сводную таблицу, в которой можно изменять начальные параметры и получить соответствующие им результаты. Начальные параметры определяются предварительно при моделировании всей параметрической схемы.

## II. УПРАВЛЕНИЕ ПАРАМЕТРАМИ

Массовые и инерционные характеристики для тел корпуса и исследуемого тела рассчитываются из их геометрии и свойств материалов. Упругая связь является имитацией соответствующего виброгасителя, в которой используются только его упругие свойства (жесткость в заданном направлении).

Пользователь задаёт параметризованные исходные данные для начального варианта проекта, переменные проекта, переменные состояния вместе с их предельными значениями и целевую функцию. В процессе оптимизации выбираются новые значения переменных модели, анализируется новый вариант модели, оцениваются переменные состояния, а затем результаты используются для повторения всей последовательности действий для достижения экстремума целевой функции. На рис. 2 представлено содержимое блока управления со сводной таблицей, формируемой в данном методе решения.

В блоке входных параметров осуществлена возможность изменения нормальной жесткости

(вдоль линии связи между элементами) упругой связи. В блоке выводимых результатов отображены соответствующие значения виброскоростей и виброускорений по трём направлениям декартовой системы координат для точки исследуемого тела.

## III. ЗАКЛЮЧЕНИЕ

Использование параметрических схем при моделировании виброустойчивых синтезаторов частот СВЧ-диапазона возможно при наличии хорошо коррелируемых между собой конструктивных параметров виброизоляторов, материалов и PSD воздействующей случайной вибрации. Комплексное применение виброизоляторов в конструкциях синтезаторов частот позволяет определить наиболее оптимальные характеристики системы виброзащиты и обеспечить требуемый уровень устойчивости конструкции к заданному уровню вибрации.

## IV. СПИСОК ЛИТЕРАТУРЫ

1. A. Apte, U.L. Rohde, A. Poddar, M. Rudolph Optimizing phase-noise performance: theory and design techniques for a crystal oscillator // IEEE Microwave Magazine. – vol. 18, nr. 4. – 2017.– P.108 – 123.
2. B. Kuran and B. Karaman Robust design studies for vibration and shock isolation of sensitive electronic equipments // IEEE 8th International Conference on Recent Advances in Space Technologies, Istanbul, Turkey. 2017. – P.17-22.
3. B.Y. Qiang and X. Deng-feng Modeling and numerical simulations of an active vibration isolation module by the power flow analysis // 3rd IEEE World Conference on Mechanical Engineering and Intelligent Manufacturing, Shanghai, China, 2020. – P.546-551.
4. D.E. Newland An introduction to random vibrations, spectral and wavelet Analysis. – Dover Publications, 2013. – 512 pp.

# ИНТЕРПРЕТАЦИЯ НЕКОТОРЫХ АСПЕКТОВ ТЕОРИИ ВЕРОЯТНОСТИ ПРИ РАЗРАБОТКЕ КОМПЬЮТЕРНЫХ ИГР

Коршикова Д. В., Гуревич О. В., Кукин Д. П., Шатилова О. О.

Кафедра вычислительных методов и программирования,

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: korshikova@bsuir.by, o.gurevich@bsuir.by, o.shatilova@bsuir.by, kudin@bsuir.by

*В данной статье рассматриваются некоторые положения теории вероятности для применения их в сфере разработки компьютерных игр. Представлены специфические трактовки аспектов теории вероятности для расчета шансов и других вероятностных событий в игре.*

## ВВЕДЕНИЕ

В играх часто используются случайности: где-то от этого зависит вероятность попадания, а где-то, если это, например, игра с костями, бросок может определять количество шагов за текущий ход.

Однако теория вероятности в играх часто работает не так, как ожидается. Самый простой пример — подбрасывание монетки. Если орёл выпал уже три раза подряд, то человек будет ожидать, что дальше будет решка. Однако предыдущие результаты никак не меняют вероятность следующих — она всё так же будет 50% для каждой из сторон. Создавая игру, разработчик сам решает, как будет работать теория вероятностей. И он может изменить ее так, чтобы помочь игрокам получить лучший опыт. Для того, чтобы это сделать каждому разработчику необходимо учесть определенные трактовки теории вероятности при разработке игр.

### I. ТРАКТОВКА «ОТ НУЛЯ ДО ЕДИНИЦЫ»

Вероятность может быть только от 0% до 100%, то есть от 0 до 1, не больше и не меньше. Можно сказать, что что-то случится с вероятностью 10%, но такой вероятности как 10% или 110% нет. 0% вероятности события означает, что это событие не произойдет, 100% — это определено случится. Например, если бросить кость семь раз, вероятно, что шестерка все-таки выпадет один раз (на самом деле, шанс равен приблизительно 72%). Когда при подсчете вероятности получается число больше, чем 100% (или меньше, чем 0%), можно утверждать, что произошла ошибка в расчетах.

### II. ТРАКТОВКА: «ЖЕЛАНИЕ, РАЗДЕЛЕННОЕ НА ВОЗМОЖНЫЕ РЕЗУЛЬТАТЫ РАВНО ВЕРОЯТНОСТИ»

Для получения вероятности чего-либо необходимо взять количество «желаемых» результатов и поделить его на количество возможных результатов (при условии, что результаты равновероятны). Например, шанс получить шестерку при броске кубика равен  $1/6$  или около 17%

(имеется шесть возможных результатов и один желаемый). А шанс, что выпадет парное число равен  $3/6$  или 50%, так как на кубике 3 парных числа.

### III. ТРАКТОВКА: «ПЕРЕЧИСЛЕНИЕ КАК СПОСОБ ПОИСКА РЕШЕНИЯ»

В большинстве случаев рассчитать вероятность того или иного события в игре бывает довольно сложно. Причина кроется в том, что те два числа, которые нам нужны (число «желаемых» результатов и число ожидаемых результатов), не всегда бывают очевидными. Например, для того чтобы подсчитать каким будет шанс выпадения, по крайней мере, двух «орлов» при трех попытках подбрасывания монеты необходимо определить число «желаемых» результатов. Самый простой способ — перечислить все возможные результаты: 1 — ООО; 2 — ООР; 3 — ОРО; 4 — ОРР; 5 — РОО; 6 — РОР; 7 — РРО; 8 — РРР. В итоге получилось восемь возможных результатов из них четыре раза «орел» выпадает, по крайней мере, дважды. Соответственно, шанс будет равен  $4/8$  или 50%. Подобный подсчет поможет решить любую проблему, связанную с вероятностью, если на это есть время. Перечисление может быть очень удобным подходом, но, если оно занимает слишком много времени, нужно искать более короткие пути.

### IV. ТРАКТОВКА: «В НЕКОТОРЫХ СЛУЧАЯХ «ИЛИ» ОЗНАЧАЕТ СЛОЖЕНИЕ»

Когда необходимо определить шанс того или иного события и при условии, что эти два события являются взаимоисключающими (они оба не могут произойти одновременно), нужно сложить их индивидуальные вероятности, чтобы получить общую вероятность. Например, чтобы подсчитать с какой вероятностью из колоды карт вытянут фигурную карту или туза, необходимо определить шанс, с которым вытянут фигурную карту  $-12/52$  и шанс вытянуть туза  $-4/52$ . Поскольку эти события взаимоисключающие, их можно суммировать:  $12/52 + 4/52 = 16/52$ , или 31% вероятности. Система сложения вероятностей может быть весьма полезной, но

только если есть уверенность в том, что события являются взаимоисключающими.

#### V. ТРАКТОВКА: «В НЕКОТОРЫХ СЛУЧАЯХ «И» ОЗНАЧАЕТ УМНОЖЕНИЕ»

Эта трактовка практически является противоположной предыдущей. Если необходимо узнать, чему равняется вероятность двух событий, которые происходят одновременно, необходимо умножить их вероятности, но только если эти два события не взаимоисключающие. Для того чтобы узнать вероятность выпадения двух шестерок при броске двух игральных костей, необходимо умножить вероятность двух событий: шанс получить шесть на одной кости равняется  $1/6$ , а также шанс получить 6 на второй кости, который тоже равняется  $1/6$ . Получается, что шанс выпадения двух шестерок  $- 1/6 \times 1/6 = 1/36$ . Возможно одинаково успешно прийти к этому выводу путем перечисления, но это отнимает намного больше времени.

#### VI. ТРАКТОВКА: «ОДИН МИНУС ДА = НЕТ»

Такая трактовка является скорее интуитивной. Если шанс какого-то события равен 10%, то шанс, что это событие не произойдет — 90%. Иногда вычислить вероятность происшествя какого-то события сложнее, чем вероятность того, что оно не произойдет.

Например, шанс, что двойная шестерка выпадет после одного броска, равен  $1/36$ . Таким образом, вероятность того, что двойная шестерка не выпадет —  $1 - 1/36$  или  $35/36$ .

#### VII. ТРАКТОВКА: «СУММА НЕСКОЛЬКИХ ЛИНЕЙНЫХ СЛУЧАЙНЫХ ВЫБОРОВ — ЭТО НЕ ЛИНЕЙНЫЙ СЛУЧАЙНЫЙ ВЫБОР»

Линейный случайный выбор — это просто случайное событие, в котором все результаты имеют одинаковую вероятность. Бросание игральной кости — это отличный пример линейного случайного выбора. Хотя, если бросить несколько игральных костей, то возможные результаты не будут иметь одинаковую вероятность. Например, если бросить две кости, то шанс получить семь довольно высок, в то время как шанс получить 12 намного меньше. Если рассмотреть все возможные варианты получения суммы игральных костей бросая две кости и построить диаграмму шансов получения каждого из результатов (от 2 до 12), то можно увидеть, что шанс получения 7 ( $1+6$ ,  $2+5$ ,  $3+4$ ) значительно выше, чем для 12 (только  $6+6$ ). По такой диаграмме можно построить кривую распределения вероятности, по которой можно оценить шансы появления каждого из результатов.

Геймдизайнеры, которые собираются использовать механику шанса как инструмент для создания своих игр, должны понимать, какая кривая распределения вероятности нужна именно им, а также знать, как ее получить.

#### VIII. ПРАКТИЧЕСКАЯ ВЕРОЯТНОСТЬ

Все рассмотренные трактовки базировались на основе теоретических вероятностях, иными словами, тем, что должно случиться. Существует также практическая вероятность, которая является меркой того, что уже случилось. Например, если бросить кость, теоретическая вероятность выпадения шестерки составит ровно  $1/6$  или 16.67%. Но можно вычислить и практическую вероятность, бросив игральную кость 100 раз и записав, сколько раз попадались шестерки. Могут выпасть 20 шестерок из 100. В этом случае практическая вероятность составит 20%, что не слишком сильно отличается от теоретической вероятности. Конечно, с увеличением количества попыток, практическая вероятность все ближе приближается к теоретической. Это правило получило название метод «Монте-Карло» в честь знаменитого казино. Положительной чертой использования метода Монте-Карло для вероятности является то, что он не требует сложных математических подсчетов — вы проделываете одно и то же действие много раз и просто записываете результаты. Иногда результаты подобных тестов могут быть полезнее теоретической вероятности. Если существуют факторы, которые нельзя учесть при математических подсчетах (кубик немного склоняется к шестерке) или же эти подсчеты настолько сложные, что нельзя составить теоретическую картину ситуации, метод Монте-Карло — это то, что поможет спрогнозировать вероятность того или иного события в игре.

#### IX. ЗАКЛЮЧЕНИЕ

Несмотря на то что геймдизайн — это творческое занятие, в нём очень часто нужно применять математику. В основном это статистика и теория вероятности. Подобная наука не определяет точного результата игры, а лишь дает оценку возможностям и шансам игроков.

#### X. СПИСОК ЛИТЕРАТУРЫ

1. Jesse Shell The art of game design / J. Shell, — 2019. —640с.
2. Вероятности и случайность в полнейшем беспорядке [Электронный ресурс] / Вероятности и случайность в полнейшем беспорядке. —Режим доступа: <http://aushestov.ru> — Дата доступа: 29.10.2021.

# МЕТОД ОТЛОЖЕННОГО ЗАТЕНЕНИЯ ТРЁХМЕРНЫХ СЦЕН, ИСПОЛЬЗУЮЩИЙ БУФЕР ВИДИМОСТИ

Красковский П. Н., Серебряная Л. В.

Кафедра программного обеспечения информационных технологий  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: kraskovskij@bsuir.by

*В данной работе рассмотрены основные подходы к затенению трёхмерных сцен. Представлен метод отложенного затенения, использующий буфер видимости. Проанализированы достоинства и недостатки рассмотренных методов, а также их эффективность.*

## ВВЕДЕНИЕ

С увеличением мощности компьютерного оборудования стало возможным использование физически достоверных методов освещения в программах трёхмерной визуализации и видеоиграх. Такие методы требуют больших вычислительных ресурсов для вывода изображения на современные дисплеи, имеющие высокое разрешение.

При использовании наивных методов затенения в процессе визуализации трёхмерной сцены освещение для каждого пикселя может быть пересчитано несколько раз, что приводит к уменьшению скорости генерации изображения. Падение производительности особенно заметно при большом количестве источников света.

Чтобы уменьшить количество вычислений, необходимо использовать методы отложенного затенения, при которых расчёт освещения происходит после определения атрибутов всех видимых фрагментов сцены.

## I. МЕТОД ПРЯМОГО ЗАТЕНЕНИЯ

Простым способом визуализации трёхмерных сцен является метод прямого затенения, при котором все расчёты освещения происходят во время заполнения Z-буфера.

Z-буфер представляет собой двумерный массив данных, который хранит глубину нарисованных фрагментов сцены. Графически Z-буфер можно представить в виде полуплоского изображения, каждый пиксель которого имеет определённый оттенок серого (см. рис. 1).

Перед рисованием очередного фрагмента необходимо сравнить значение глубины этого фрагмента со значением глубины нарисованного в той же позиции фрагмента в Z-буфере. Если значение глубины нового фрагмента меньше значения глубины в Z-буфере (фрагмент расположен ближе к наблюдателю), необходимо заменить значение глубины в Z-буфере на значение глубины нового фрагмента. В противном случае новый фрагмент не рисуется [1].

Очевидно, что количество замен значений в Z-буфере, а также количество вычислений освещения зависит от порядка рисования фрагмен-

тов. Если первыми рисуются самые близкие к наблюдателю фрагменты, количество вычислений будет наименьшим, так как дальние фрагменты не пройдут проверку Z-буфера и не будут нарисованы. Наихудший случай – когда фрагменты рисуются в обратном порядке от самых дальних к самым близким. В таком случае количество вычислений освещения будет наибольшим.

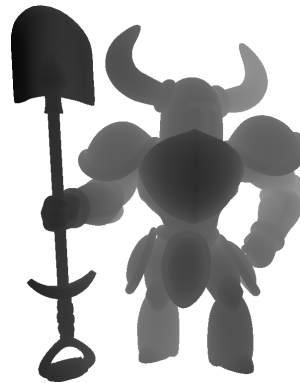


Рис. 1 – Графическое представление Z-буфера

## II. МЕТОД ОТЛОЖЕННОГО ЗАТЕНЕНИЯ

Чтобы избавиться от избыточных вычислений освещения, свойственных методу прямого затенения, необходимо использовать метод отложенного затенения [2]. Основная идея данного метода заключается в том, чтобы перенести все тяжёлые вычисления освещения на более поздний этап.

Визуализация трёхмерной сцены методом отложенного затенения состоит из двух шагов. На первом шаге определяются атрибуты всех видимых фрагментов сцены, которые хранятся в виде коллекции текстур, называемой G-буфером. Данный буфер содержит информацию о позициях, нормалях, цветах и других характеристиках визуализируемых объектов (см. рис. 2). В отличие от вычислений освещения формирование G-буфера не требует больших вычислительных затрат.

На втором шаге метода выполняются вычисления освещения с использованием данных, записанных в G-буфер. При этом вычисления

производятся только для тех фрагментов, которые видны наблюдателю. Все невидимые фрагменты отбраковываются на предыдущем шаге во время заполнения Z-буфера.

Очевидным недостатком данного метода является повышенное потребление оперативной памяти, так как в G-буфере требуется сохранять большое количество текстур. Также данный метод не подходит для визуализации простых сцен с малым количеством объектов и источников света и может работать медленнее метода прямого затенения.



Рис. 2 – Содержимое G-буфера: позиции, нормали, альbedo, металличность, шероховатость

### III. БУФЕР ВИДИМОСТИ

Потребление оперативной памяти можно уменьшить, если вместо G-буфера использовать буфер видимости, который хранит индексы объектов и треугольников, которым принадлежат видимые фрагменты [3]. Тогда на втором шаге метода по этим индексам можно получить атрибуты вершин треугольников для нахождения атрибутов соответствующих фрагментов, используя барицентрические координаты.

Данный метод позволяет существенно сократить потребление оперативной памяти, а также подходит для визуализации сцен любой сложности. Тем не менее метод не подходит для визуализации полупрозрачных объектов, так как в буфере содержится информация только о самых близких к наблюдателю фрагментах. Данная проблема свойственна всем методам отложенного затенения, но отсутствует у метода прямого затенения.

### IV. АНАЛИЗ ЭФФЕКТИВНОСТИ МЕТОДОВ

На рисунках 3 и 4 представлены диаграммы сравнения скорости работы и потребления оперативной памяти рассмотренными ранее методами. Все замеры производились для трёх сцен, состоящих из одного объекта и одного источника света. В каждой сцене объекты отличались геометрической сложностью и количеством многоугольников. Все вычисления производились на процессоре без использования графических API. Изоб-

ражение выводилось на дисплей с разрешением  $1366 \times 768$ .

На рисунках видно, что наилучшее соотношение скорости и потребления оперативной памяти показывает метод, использующий буфер видимости. Прирост скорости особенно заметен на объектах, имеющих сложную геометрию и большое количество многоугольников. Худшие показатели у классического метода отложенного затенения. Это обусловлено описанными ранее недостатками данного метода – замедление скорости работы при малом количестве объектов и источников света, а также повышенное потребление оперативной памяти.

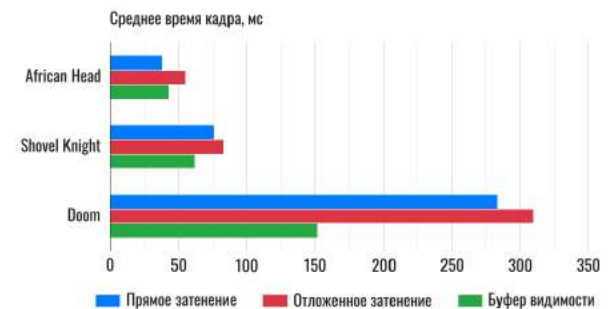


Рис. 3 – Сравнение скорости работы различных методов затенения

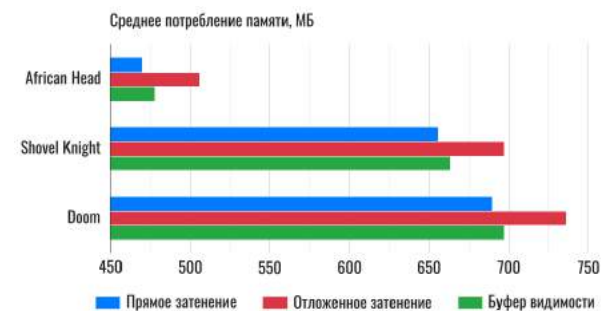


Рис. 4 – Сравнение потребления памяти различных методов затенения

### V. ВЫВОДЫ

Метод отложенного затенения, использующий буфер видимости, целесообразно использовать при визуализации трёхмерных сцен любой сложности. Если в сцене присутствуют полупрозрачные объекты, необходимо использовать дополнительные механизмы для правильного отображения этих объектов при использовании данного метода.

1. Real-Time Rendering / T. Akenine-Möller [et al.] – 4th ed. – Boca Raton : Taylor & Francis, CRC Press, 2018. – 1198 p.
2. The triangle processor and normal vector shader: a VLSI system for high performance graphics / M. Deering [et al.] – New York : Association for Computing Machinery, 1988. – P. 21–30. – (ACM SIGGRAPH Computer Graphics ; 22(4)).
3. Burns C. A. The Visibility Buffer: A Cache-Friendly Approach to Deferred Shading / C. A. Burns, W. A. Hunt. – P. 55–69. – (Journal of Computer Graphics Techniques (JCGT) ; 2(2))

# НЕЙРОСЕТЕВЫЕ АЛГОРИТМЫ КАДРОВОГО ПЛАНИРОВАНИЯ

Никульшина К. Б., Гуринович А. Б.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Кафедра вычислительных методов и программирования,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: loveapple15112005, gurinovich@bsuir.by

*Исследуются нейросетевые алгоритмы для эффективной организации кадрового планирования. Показано, что их применение является составной частью рационального управления в ИТ-отрасли*

## ВВЕДЕНИЕ

Нейронная сеть — это математическая модель для воспроизведения работы человеческого мозга. Нейросетевые модели используются в задачах прогнозирования, для распознавания образов. Особую эффективность они показали в задачах управления.

Искусственная нейронная сеть обучается с учителем. Это означает наличие обучающего набора (датасета), который содержит примеры с истинными значениями: тегами, классами, показателями. Возможность обучения — главное преимущество нейронных сетей. Технически обучение заключается в нахождении коэффициентов связей между нейронами. В процессе обучения нейронная сеть выявляет сложные зависимости между входными данными и выходными и выполняет обобщение.

## I. НЕЙРОСЕТЕВЫЕ АЛГОРИТМЫ В УПРАВЛЕНИИ

Нейросетевые алгоритмы эффективны для планирования бизнес-процессов и кадрового планирования, в частности. При прогнозировании возможности увольнения датасетом будет список факторов, которые могут повлиять на увольнение (частота командировок, отсутствие премии и роста квалификации персонала и многие другие).

Нейронные сети можно использовать для:

- прогноза на основе наблюдаемых данных обучения;
- классификации невидимых данных в заранее определенные группы на основе характеристик, наблюдаемых в тренировочных данных;
- группировки обучающих данных в естественные группы на основе сходства характеристик в тренировочных данных.

Но есть и минусы, которые были выявлены при внедрении, а именно:

1. Для внедрения нейросетей нужны свободные деньги. Аренда инфраструктуры, зарплата разработчиков, перестройка бизнес-

процессов требует больших финансовых вложений.

2. Для успешного внедрения нейросетей мало выделить 1 млн долларов и найти разработчиков высокого класса. Необходимо делать A/B-тесты, нанять аналитика и убедить команду пользоваться инновациями.
3. Нехватка программистов. Спрос на программистов во всем мире сейчас превышает предложение.
4. Нехватка времени. Сильная команда разработчиков может за месяц создать с «нуля» и ввести в работу нейросеть. Но такие «сокровища» есть только у корпораций.

## II. ИСПОЛЬЗОВАНИЕ НЕЙРОСЕТЕВЫХ АЛГОРИТМОВ В КАДРОВОМ ПЛАНИРОВАНИИ В ИТ-ОТРАСЛИ

В HR (Human Resource) — подразделениях нейронные сети занимаются аналитикой кадрового потенциала. В 86% случаев они дают точный прогноз.

Основные блоки использования:

1. Оптимизирует поиск подходящих кандидатов на открытую в компании должность.
2. Оценивает навыки сотрудников и подбирает лучшие решения для обучения персонала.
3. Упрощает формирование обоснованной системы оплаты труда.
4. Помогает управлять мотивацией и эффективностью труда сотрудников

Нейросети анализируют тысячи признаков и способны вовремя указать на «выгоревших» сотрудников. Алгоритмы прогнозируют возможные сценарии увольнения персонала, помогают вовремя скорректировать условия труда и разработать эффективную систему мотивации.

Многие крупные бизнесы вкладывают в создание интеллектуальных ассистентов, корпоративных мессенджеров и приложений. Удобный сервис оформления командировок экономит несколько часов.

HR-отделы Vodafone, Intel, Nike используют нейросети, чтобы заранее предсказать увольнение менеджера. Соискатели могут использо-

вать нейросети. Приложение загружает видео с презентацией, и анализирует поведение, жесты и мимику соискателя — и кадровый агент направляет вам рекомендации об изменениях параметров и характеристик, необходимых для потенциальных работодателей. Использование алгоритмизации в кадровом планировании минимизирует ненужные коммуникации, создает комфортную рабочую среду.

На руководителя возлагается контроль алгоритмов, принятие решений, корректировка бизнес-процессов. Исполнители освобождаются от рутинных и трудоемких операций.

Прогнозирование в сфере HR связано с внедрением технологий машинного обучения. Метод изучает и анализирует экспериментальные данные. Он выявляет неочевидные закономерности в опытных данных. Это помогает выстроить кадровую политику, организовать системную профилактику потерь. Прогнозы стимулируют качественные изменения в корпоративной культуре, программе мотивации, обучении и развитии персонала. Для эффективной прогнозной аналитики потребуется наличие двух условий. Большие данные. В большинстве компаний их уже накоплено много с помощью активных и пассивных источников информации. Активные источники включают обратную связь от сотрудников посредством анкет, пульс-опросов; пассивные — сбор данных из CRM, электронной почты, мессенджеров, соцсетей и т.д.

Компания Google с помощью предсказательной аналитики удерживает таланты. Если искусственный интеллект сигнализирует, что ключевой сотрудник подумывает об уходе, его повышают или предлагают сменить локацию и попробовать себя в другом бизнес-направлении.

Регулярные исследования помогли вывести формулу сохранения команды в sales-подразделении. Менеджеры по продажам, не получившие повышение за 4 года, чаще сидят на рабочих сайтах и ищут нового работодателя, чем трудятся над увеличением продуктивности.

Для прогнозирования текучести персонала в предиктивной аналитике существует ряд методов. Логистическая регрессия — самый распространенный из них. Она предсказывает вероятность ухода сотрудника, раскрывает причины увольнения и даёт рекомендации по снижению уровня текучести. Индийская компания HBSC — пример успешного применения этого метода. HBSC разработала аналитическую модель, кото-

рая предсказывает вероятность потери сотрудника. В качестве вводных использовали демографические данные, полученные в результате исследований. Это позволило организации решить проблемы с текучестью кадров.

### III. ЗАКЛЮЧЕНИЕ

Исследуются перспективы использования нейросетевых алгоритмов для управления персоналом в ИТ-бизнесе. Показано, что внедрение нейросетевых технологий в решение задачи по управлению персоналом эффективно на всех этапах: прогнозирования, планирования и управления.

### IV. СПИСОК ЛИТЕРАТУРЫ

1. Николенко, С. Глубокое обучение: Погружение в мир нейронных сетей / С. Николенко, А. Кадури, Е. Архангельская // Санкт-Петербург -- Глубокое обучение. -- СПб: Питер, 2018. -- 480 с.
2. Предиктивная аналитика в HR: модно или просто-напросто необходимо [Электронный ресурс] / Тренинговый портал Беларуси «Аспект». – Режим доступа: <https://ta-aspect.by/articles/prediktivnaya-analitika-v-hr-modno-ili-prosto-naprost-neobходимо>. – Дата доступа: 25.10.2021.
3. ПГлава 2. Введение [Электронный ресурс]. – Режим доступа: <https://neuralnet.info/chapter/введение/>. – Дата доступа: 25.10.2021.
4. «Главным сотрудником компании станет искусственный интеллект» [Электронный ресурс] / Laba. – Режим доступа: <https://l-a-b-a.com/blog/566-digital-v-hr-avtomatizacija-kadrovyykh-processov>. – Дата доступа: 25.10.2021.
5. Предиктивная аналитика в HR: сферы применения [Электронный ресурс] / Yva.AI. – Режим доступа: <https://www.yva.ai/ru/blog/predskazatelnaya-prediktivnaya-analitika-v-hr-sfery-primeniya>. – Дата доступа: 25.10.2021.
6. Нейронные сети — математический аппарат [Электронный ресурс] / Loginom Company ООО «Аналитические технологии». – Режим доступа: <https://basegroup.ru/community/articles/math>. – Дата доступа: 25.10.2021.
7. Как работает нейронная сеть: алгоритмы, обучение, функции активации и потери [Электронный ресурс]. – Режим доступа: <https://neurohive.io/ru/osnovy-data-science/osnovy-nejronnyh-setej-algoritmy-obuchenie-funkcii-aktivacii-i-poteri/>. – Дата доступа: 25.10.2021.
8. Explained: Neural networks [Электронный ресурс] / Massachusetts Institute of Technology. – Режим доступа: <https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414>. – Дата доступа: 25.10.2021.
9. Neural Networks and Deep Learning [Электронный ресурс]. – Режим доступа: <http://neuralnetworksanddeeplearning.com>. – Дата доступа: 25.10.2021.

# ГЕНЕТИЧЕСКИЙ АЛГОРИТМ. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ЭТАПА МУТАЦИИ

Носкова Е. Е., Дружинская Е. В.

Уфимский государственный нефтяной технический университет

Уфа, Россия

E-mail: noskova1610@gmail.com, alena1806@mail.ru

*В статье представлен краткий обзор генетического алгоритма, и программная реализация его этапа мутации. Статья посвящена комплексному исследованию этапа мутации генетических алгоритмов. Произведен сбор и анализ научной литературы, проведены систематизация и структурирование сведений о генетическом алгоритме, особое внимание было уделено этапу мутации: выделены и описаны основные алгоритмы изменения последовательности на данном этапе и критерии окончания цикла, представлен словесно-описательный алгоритм и результаты его программной реализации. Кратко даны определения понятий, используемых в генетических алгоритмах, рассмотрены операторы работы с данными алгоритмами, построена обобщённая модель генетического алгоритма в виде блок-схемы. Реализация позволяет продемонстрировать использование этапа мутации в генетическом алгоритме.*

## ВВЕДЕНИЕ

Естественный отбор является основным фактором эволюции. Суть естественного отбора заключается в том, что в результате действий в популяции увеличивается число наиболее приспособленных особей, так как они имеют больше шансов на выживание и размножение. Потомки получают лучшие качества от родителей вследствие генетического наследования. Но в природе не всегда побеждает сильнейший из-за нерегулируемых случайностей. В отличие от природных явлений, программно реализованные генетические алгоритмы предоставляют возможность принудительного выбора и сохранения лучших экземпляров набора. Основная идея генетических алгоритмов заключается в том, что с использованием принципов «естественного отбора» среди пробных решений выбирается наиболее качественное. Генетические алгоритмы дают возможность оптимизировать функции дискретных переменных, непрерывные функции, компьютерные программы, а также, принимать решение о стратегии выполнения активности для достижения поставленной цели и сокращать перебор в сложных задачах. Алгоритм позволяет сократить число переборов при нахождении значений функции, описывающей сложные задачи. А также, позволяет в условиях неопределённости выполнять подбор наиболее правильного решения с максимальным соблюдением всех ограничений, накладываемых на результат.

Генетические алгоритмы применяются для настройки и обучения искусственной нейронной сети, задач компоновки, составления расписаний и игровых стратегиях, а также, робототехнике и создание моделей искусственной жизни.

## I. ГЕНЕТИЧЕСКИЙ АЛГОРИТМ

Алгоритм работает на сгенерированной начальной популяции – некотором массиве случайных объектов, возможно, но не обязательно при-

ближенных к решению. Выборка производится из представленного набора данных, и все операции применяются к выборке до достижения поставленных ограничений [1].

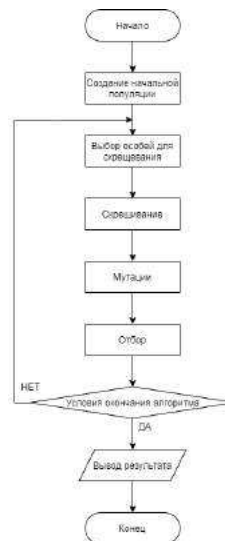


Рис. 1 – Графическая модель алгоритма

## II. ОПЕРАТОРЫ ГЕНЕТИЧЕСКОГО АЛГОРИТМА

Реализация и эффективность генетических алгоритмов зависит от выбора операторов. Основными операторами генетического алгоритма являются мутация, отбор (селекция), скрещивание и выбор родителей [4].

Рассмотрены наиболее распространённые операторы выбора пары, которые станут родителями:

Панимиксия реализует случайное бинарное отношение. При этом некоторые элементы могут образовать несколько пар, а какие-то не участвовать в данном отношении. Поскольку с каждым применением оператора популяция увеличивается, то эффективность алгоритма падает за счет зашумленности данных.



Инбридинг – метод, при котором оба родителя являются членами одной популяции, и первый родитель выбирается случайным образом, а второй родитель является ближайшим «родственником» к первому [3].

При аутбридинге используется понятие схожести особей, в котором брачные пары формируются из максимально далеких по родству особей [1].

Оператором скрещивания выступает рекомбинация, которая применяется сразу же после отбора предков для получения нового поколения. Смысл воспроизведения заключается в том, что потомки наследуют генную информацию от обоих родителей.

### III. МУТАЦИИ

Нахождение решений внутри замкнутой системы приводит к ее энтропии.

Этап мутации необходим для вывода популяции из локального экстремума. Изменение случайно выбранного элемента в объекте препятствует преждевременной сходимости популяции к неправильному решению. Следующие операторы мутации изменяют состав объекта:

1. присоединение случайного к концу последовательности;
2. вставка случайного гена в случайно выбранную позицию в последовательности;
3. удаление случайно выбранного гена из последовательности;
4. обмен местами в последовательности двух соседей одного случайно выбранного гена.

В программной реализации случайным образом генерируется последовательность, после чего, происходит изменение последовательности в соответствии с выбранным оператором мутации. Реализованы и запущены все 4 оператора. На вход поступает последовательность генов, после чего происходит изменение в составе объекта, которые и обозначает мутацию. На выходе имеем последовательность с тем же количеством генов, измененную по некоторому правилу, изменяющим популяцию.

Последовательность задается массивом чисел, который генерируется случайным образом используя функцию `rand()` и аргумент `Time (NULL)`, позволяющий генерировать неповторяющуюся последовательность генов, используя текущее время. После этого данные массива отправляем в функцию, в которой используя случайные числа генерируется элементы для изменения структуры объекта.

```

0 1 1 1 1 1 0
Присоединение к концу последовательности
1 1 1 1 1 1 0
Вставка случайного гена из совокупности всевозможных значений генов в случайно выбранную позицию
0
0 1 1 1 1 1 0
Удаление случайно выбранного гена из последовательности
2
0 1 1 1 1 1 0
Обмен местами в последовательности двух соседей одного случайно выбранного гена
2
0 1 1 1 1 1 0

```

Рис. 2 – Программная реализация этапа мутации после первой итерации

```

1 0 1 1 0 1 1
Присоединение к концу последовательности
0 1 1 0 1 1 1
Вставка случайного гена из совокупности всевозможных значений генов в случайно выбранную позицию
5
0 0 1 1 1 1 1
Удаление случайно выбранного гена из последовательности
2
1 1 1 0 1 1 1
Обмен местами в последовательности двух соседей одного случайно выбранного гена
2
0 1 1 1 0 1 1

```

Рис. 3 – Программная реализация этапа мутации после второй итерации

Для создания маркированного списка нужно использовать команду `\List`, в которой каждый пункт списка обозначается тегом `\item`.

### IV. ВЫБОРКА НОВОГО ПОКОЛЕНИЯ

На этапе построения выборки из нового поколения применимы несколько типов операторов. Отбор усечением использует целевую функцию, проверяющую каждую особь на пригодность к решению. Ранжирование популяции производится по убыванию значения целевой функции. Этот метод не имеет аналогов в естественной эволюции и обычно используется для больших популяций [2]. В элитарном отборе участвуют как родители, так и потомки, которые создают промежуточную популяцию. А затем, члены этой популяции оцениваются и объекты с высшей оценкой образуют следующее поколение. В отборе вытеснением выбор особи в новое поколение зависит от величины пригодности и хромосомного набора. Из всех особей с одинаковой пригодностью и приспособленностью выбирают особи с разными генотипами, таким образом поддерживается структурное разнообразие объектов.

Вновь построенная выборка проверяется на соответствие ограничивающих условий для решения. При соответствии алгоритм возвращает результат – полученную выборку. Иначе выборка рассматривается как начальная, и алгоритм запускает новую итерацию [1].

Применение генетического алгоритма позволяет в условиях неопределённости выполнять подбор наиболее правильного решения с максимальным соблюдением всех ограничений, накладываемых на результат.

### V. СПИСОК ЛИТЕРАТУРЫ

1. Панченко, Т. В. Генетические алгоритмы / Ю. Ю. Тарасевич // Астрахань: Издательский дом «Астраханский университет» – 2007. – С. 57-65.
2. Модификации генетических алгоритмов [Электронный ресурс] / . – Режим доступа: <https://intuit.ru/studies/courses/14227/1284/lecture/24172?page=2> – Дата доступа: 25.09.2021.
3. Гладков, Л. А., Курейчик, В. В., Курейчик В. М. Генетические алгоритмы / В. М. Курейчик // М.: ФИЗМАТЛИТ. – 2010. – С. 124–137.
4. Генетический алгоритм. Просто о сложном [Электронный ресурс] / Режим доступа: <https://habr.com/ru/post/128704/> – Дата доступа: 10.10.2021.

# СМАРТ-СИСТЕМА ПЕРСОНАЛЬНОГО МОНИТОРИНГА СОСТОЯНИЯ ЗДОРОВЬЯ ЧЕЛОВЕКА

Осипов А. Н., Хазановский И. О., Котов Д. В., Фролов А. В., Пацев А. В., Пацев С. В.

Научно-исследовательская часть,

Белорусский государственный университет информатики и радиоэлектроники

Республиканский научно-практический центр «Кардиология»

Республиканский центр медицинской реабилитации и бальнеолечения

Городской клинический родильный дом № 2

Минск, Республика Беларусь

E-mail: osipov@bsuir.by, kotov@bsuir.by, frolov.minsk@gmail.com, paceev@yandex.ru, paceevserg@mail.ru

*Рассмотрены принципы проектирования системы внегоспитального, вневрачебного мониторинга состояния человека. Изложены медицинские методики классификации физиологического состояния человека, определены соответствующие типы сенсоров биомедицинских сигналов, методов обработки и передачи анализируемой информации и принятия решения.*

## ВВЕДЕНИЕ

Проблема охраны здоровья приобретает особую значимость для всего человечества в связи с ростом продолжительности жизни и старением населения в развитых странах, демографическим взрывом в странах третьего мира, ограниченными кадровыми и финансовыми ресурсами здравоохранения, обострением экологической ситуации и глобальной пандемией, вызванной коронавирусом SARS-CoV-2. Для решения данной проблемы правительства и представители крупного бизнеса вкладывают все большие средства в новые медицинские проекты, в том числе, реализованные на базе ИТ-технологий. Современный уровень информационно-коммуникационных технологий, основанных на технологиях 5G, анализа больших данных, искусственного интеллекта и интернета вещей предоставляет обширные возможности для развития электронного здравоохранения и, в частности, для дистанционной диагностики и мониторинга состояния органов и функциональных систем человека [1-2]. В связи с этим, создание и внедрение в медицинскую практику смарт-систем вневрачебного мониторинга состояния человека является актуальным.

### Принцип построения СМАРТ-СИСТЕМЫ МОНИТОРИНГА СОСТОЯНИЯ ЗДОРОВЬЯ

Мониторинг витальных (жизненно-важных) функций организма следует ориентировать на современные классификации функциональных состояний (интегральная характеристика жизнеспособности биологической системы на данный момент времени). С позиции физиологического подхода состояние физиологической нормы характеризуется удовлетворительной адаптацией организма к условиям внешней среды и наличием достаточных функциональных возможностей. По мере ухудшения здоровья физиологическая норма переходит в донологическое состояние с мобилизацией функциональных ресурсов организма и напряжением

регуляторных систем. Далее по мере истощения резервов регуляторных систем следует переход в преморбидное состояние (предболезнь), проявляющееся сниженными функциональными возможностями и неудовлетворительными адаптационными способностями. Данные состояния являются обратимыми. При срыве механизмов адаптации наступают болезненные необратимые изменения с развитием специфических патологических изменений на органно-системном уровне. С клинической точки зрения выделяют: условное здоровье, функциональные отклонения, пограничные состояния, хронические заболевания, стойкое ограничение жизнедеятельности (инвалидность), полную утрату функций и смерть.

Для планирования и оказания медицинской помощи важно выделять хронические состояния (артериальная гипертензия, ишемическая болезнь сердца, сердечная недостаточность, обструктивная болезнь лёгких, сахарный диабет и др.), которые не требуют срочных вмешательств, и неотложные (критические) состояния (гипертонический криз, инфаркт миокарда, нарушения сердечного ритма, инсульт, сахарная кома, внезапная сердечная смерть и др.).

Естественно, что стратегической задачей мониторинга витальных параметров является идентификация предкризовых состояний и выработка тревожной сигнализации как для пациента, так и для медицинского персонала с целью принятия неотложных мер.

Следует отметить, что дисфункции и патологические состояния любой нозологии отражаются на эффективности системы транспорта O<sub>2</sub> и CO<sub>2</sub>, соответственно данные параметры в обязательном порядке должны анализироваться смарт-системой.

Для разработки смарт-системы персонального мониторинга состояния здоровья человека требуется решить следующие ключевые задачи [3-5]:

- произвести выбор и интегрировать смарт-сенсоры контроля физиологического состояния человека в смарт-систему;
- адаптировать существующие и разработать новые диагностические протоколы, алгоритмы и методики применительно к используемым информационно-коммуникационным технологиям для реализации задач самоконтроля и индивидуальной дистанционной врачебной диагностики пользователей;
- разработать методы интегральной обработки биомедицинской информации с использованием технологии BigData и нейросетевого анализа. С помощью технологии интернета вещей и облачных технологий реализовать передачу информации от смарт-сенсоров в персональное мобильное устройство по беспроводным каналам связи и последующий обмен данными с врачебным вычислительным сервером.

В отличие от простых систем мониторинга, построенных на сравнении контролируемых витальных параметров с пороговыми значениями нормы, разрабатываемая смарт-система реализует алгоритмы нейросетевого анализа витальных параметров, что позволяет контролировать резервы регуляторных систем организма. Анализируется синергичность между физиологическим статусом организма и физической активностью.

В основу проектируемой системы мониторинга положен принцип минимума сенсоров, размещаемых на теле пациента при максимуме извлекаемой биоинформации. В качестве исходной биоинформации используются фотоплетизмографический сигнал, регистрируемый с пальца руки, и данные акселерометра. Данный набор сенсоров позволяет мониторить частоту сердечных сокращений (ЧСС), сатурацию  $O_2$  ( $SpO_2$ ), последовательность RR-интервалов, напряженность регуляторных систем организма и физическую активность пациента.

Синергичность характеризуется отношением  $ЧСС/V$  или  $ЧСС/N$ , где  $V$  – скорость движения,  $N$  – мощность нагрузки. По данному отношению можно установить гиперактивность, нормаактивность, гипореактивность и ареактивность, согласно классификации A.Vein. Контролируются показатели  $SpO_2(t)$ ,  $ЧСС(t)$ ,  $RR(t)$ ,  $stress-index(t)$ ,  $ЧСС/V(t)$ , где  $t$  – индекс времени. Степень их физиологической важности ранжируется посредством весовых коэффициентов. Риск-стратификация пациента ранжируется по «принципу светофора»:

- предкризовые признаки не наблюдаются (зелёный);

- обнаружены кратковременные предкризовые признаки (жёлтый);
- обнаружены устойчивые предкризовые признаки (красный).

Специфической особенностью разрабатываемой смарт-системы мониторинга является контроль резервов регуляторных систем организма по данным динамики RR-интервалов, отражающих циклы сердечных сокращений. В вариациях RR-интервалов скрыта информация о состоянии симпатoadренальной и нейрогуморальной систем регуляции. Выделяются следующие типы резервов регуляции: резервы регуляции в норме, резервы регуляции снижены, истощение резервов регуляции, срыв регуляции. Интегральным показателем напряжённости систем регуляции служит стресс-индекс ( $stress-index$ ). Именно снижение и истощение резервов регуляции предшествует критическим состояниям организма человека, однако в данном случае витальные параметры могут всё ещё находиться в пределах возрастной нормы.

## ЗАКЛЮЧЕНИЕ

Таким образом, разрабатываемая смарт-система, основанная на современных классификациях функционального состояния человека, методах регистрации, передачи и анализа биомедицинской информации, является основой для создания нового этапа электронного здравоохранения – дистанционных инженерно-технических средств контроля физиологического состояния человека. Внедрение данной системы в медицинскую практику позволит обеспечить адресность и эффективность оказания медицинской помощи большому числу пациентов, превентивно выявить лиц с высоким риском жизненно опасных патологий, своевременно принять решение об экстренном лечении.

## СПИСОК ЛИТЕРАТУРЫ

1. Brahmaji G; Sangeeta Viswanadham; Appala Srinivasu Muttipati; Om Prakash Samantray. E-Healthcare Monitoring System using IoT with Machine Learning Approaches. IEEE 2020.
2. Md Rakinul Amin Refat, Palvinderjit Kaur, Sathia Priya Ramiah. E-Healthcare-Personalized Health Monitoring System, International Journal of Current Research and Review, 2020, 150-154p.
3. Bikash Pradhan, Saugat Bhattacharyya, Kunal Pal, IoT-Based Applications in Healthcare Devices, Journal of Healthcare Engineering, 2021.
4. Md. Milon Islam, Ashikur Rahaman, Md. Rashedul Islam, Development of Smart Healthcare Monitoring System in IoT Environment, SN Computer Science, 2020, 185p.
5. Nora Mahmoud, Shaker El-Sappagh, Samir M. Abdelrazek, Hazem M. El-Bakry; A Real-time Framework for Patient Monitoring Systems based on a Wireless Body Area Network, International Journal of Computer Applications, 2020, 12-21p.

# ПРОЕКТИРОВАНИЕ АРХИТЕКТУРЫ РАСПРЕДЕЛЕННОЙ СИСТЕМЫ УДАЛЕННОЙ ПЕЧАТИ

Пашук А. В., Гуринович А. Б., Волорова Н. А.

Кафедра информатики, кафедра вычислительных методов и программирования, кафедра информатики,  
Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {pashuk, gurinovich, volorova}@bsuir.by

*При организации документооборота в компаниях, использующих складские помещения, очень важно эффективно организовать этот процесс, чтобы максимально сократить время, необходимое для получения нужных документов, например, при комплектации новых заказов. Аналогичная ситуация наблюдается при приемке новых товаров, на которые необходимо распечатать и наклеить этикетки или другую информацию для дальнейшего перемещения товаров по складу.*

## ВВЕДЕНИЕ

Работники складов крупных компаний обычно одновременно используют несколько принтеров для печати различных видов документов. Например, для печати этикеток или другой печатной продукции, которая должна быть наклеена на товары, используются специальные принтеры, поддерживающие необходимый формат бумаги. Для оформления заказов или печати бланков договоров используются обычные или промышленные многофункциональные устройства печати.

Еще одним важным аспектом работы с документацией в крупных компаниях является тот факт, что принтеры подключены к нескольким компьютерам или серверам, что усложняет процесс печати. Сотруднику необходимо скачивать нужный документ и переходить от одного рабочего места к другому для печати документов.

В рамках данного проекта было разработано решение, позволяющее упростить процесс печати и осуществлять печать с любого печатного устройства без необходимости работать с компьютера, к которому оно подключено.

## I. ОПИСАНИЕ АРХИТЕКТУРЫ

Общая схема архитектуры системы удаленной печати приведена на рисунке 3. При проектировании использовался микросервисный подход, позволяющий декомпозировать сложную систему на ряд относительно самостоятельных компонентов. Достоинством данного подхода является гибкость и возможность масштабирования отдельных компонентов при необходимости (например, масштабирование микросервиса печати при увеличении количества запросов).

Каждый элемент системы представляет собой отдельный Docker [1] контейнер (или контейнеры) для упрощения процессов развертывания и масштабирования.

Система печати состоит из следующих компонентов:

- Кабинет пользователя;
- Клиентское API (Customers API);

- Платежное API (Payments API);
- Сервис Печати (Print Service);
- Сервис синхронизации (Sync Service).

Кабинет пользователя представляет собой интерфейс, через который сотрудники компании могут отслеживать состояние запросов на печать, получать информацию о количестве документов, отправленных на печать, оплатить подписку за сервис печати, проверить состояние оплаты и т.п. Данный компонент использует клиентское и платежное API для работы.

Клиентское и платежное API представляют собой вспомогательные компоненты системы и написаны с использованием фреймворка Flask (Python). В качестве базы данных для хранения информации данные сервисы используют PostgreSQL.

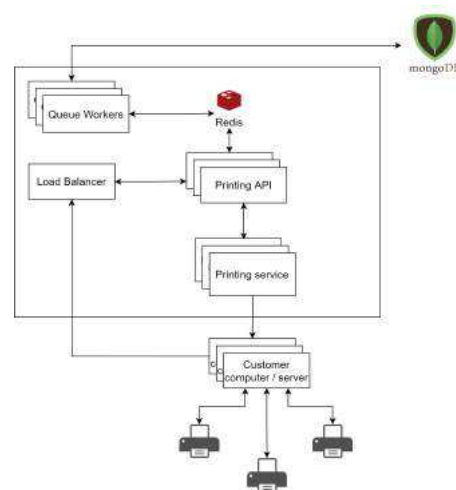


Рис. 1 – Микросервис печати

На рисунке 1 показана схема микросервиса печати, отвечающего за доставку документов на указанные принтеры. Общий принцип работы заключается в том, что на компьютеры (серверы) компании, подключенные к принтерам, устанавливается специальное клиентское приложение, которое определяет доступные принтеры и синхронизирует информацию с микросервисом печати. Приложение написано на C++ и использует

ет API операционных систем для работы с принтерами.

В систему документооборота или ERP-систему компании устанавливается интеграция, которая позволяет отправлять документы на печать без скачивания файла, а отправляя их напрямую через микросервис печати, указывая при этом принтер, на котором необходимо распечатать документ.

Микросервис печати отвечает за прием запросов на печать и отправку документов на корректные клиентские устройства, подключенные к нужным принтерам. Информация о статусе запроса и другая техническая информация хранится в базе данных MongoDB [3], что позволяет при необходимости вести статистику и оповещать техподдержку при появлении проблем.

Как было сказано выше, данный компонент системы может быть масштабирован при увеличении нагрузки.

Еще одним важным микросервисом является сервис синхронизации, позволяющий синхронизировать данные в базах данных (рисунок 2). На рисунке 3 видно, что система имеет три основных хранилища данных - две базы PostgreSQL и одну базу MongoDB. Сервис синхронизации отвечает за обновление статистики использования для каждого пользователя сервиса и другие технические задачи.

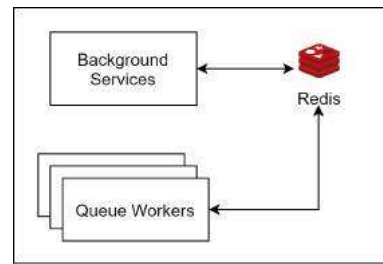


Рис. 2 – Микросервис синхронизации данных

## II. ЗАКЛЮЧЕНИЕ

В рамках данного проекта была спроектирована и разработана эффективная система удаленной печати. За время эксплуатации разработанная система показала свою отказоустойчивость за счет использования микросервисного подхода (на данный момент нагрузка составляет 3-5 тысяч документов в день).

1. Empowering App Development for Developers | Docker [Electronic resource] / Mode of access: <https://www.docker.com/>. – Date of access: 22.10.2021.
2. PostgreSQL: The world's most advanced open source database [Electronic resource] / Mode of access: <https://www.postgresql.org/>. – Date of access: 23.10.2021.
3. MongoDB: The most popular database for modern apps [Electronic resource] / Mode of access: <https://www.mongodb.com/>. – Date of access: 23.10.2021.

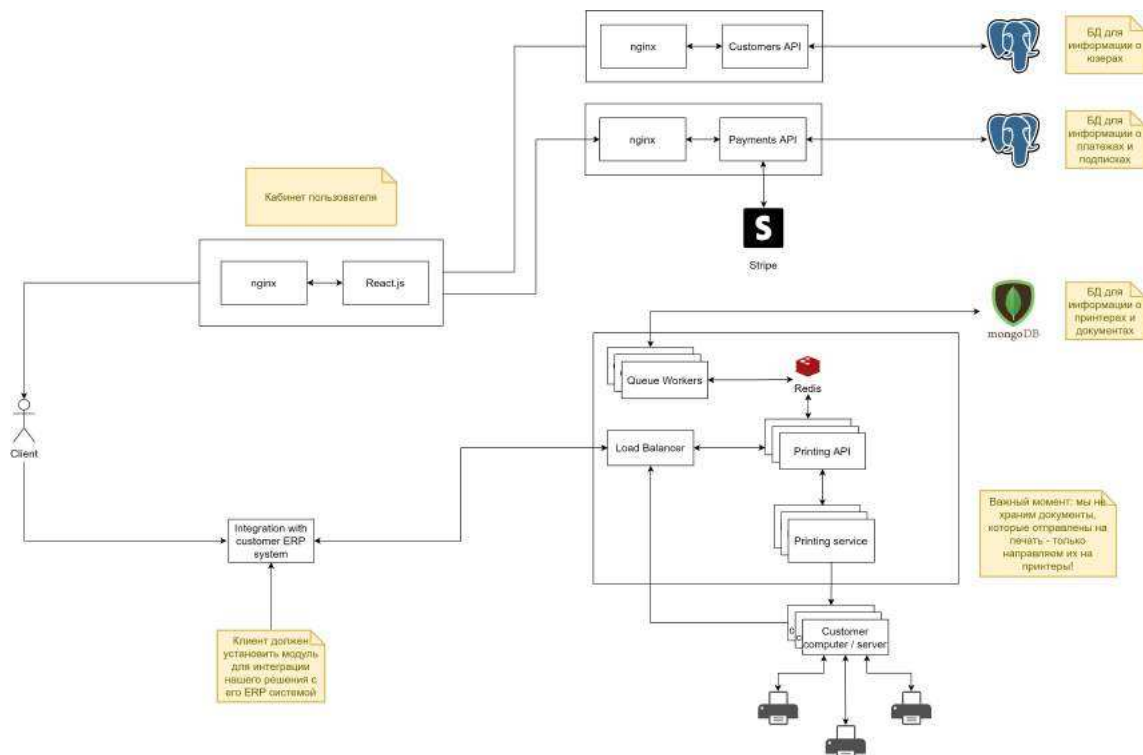


Рис. 3 – Архитектура разрабатываемой системы печати

# О ПРОБЛЕМЕ ОЦЕНКИ ОДНОРОДНОГО ПОТОКА В ДВУНАПРАВЛЕННОЙ СЕТИ

Пилипчук Л. А., Романчук М. П.

Кафедра компьютерных технологий и систем, Белорусский государственный университет

Минск, Республика Беларусь

E-mail: {pilipchuk, frm.romanchuk}@bsu.by

*Рассматривается задача локализации специальных программируемых устройств (сенсоров) в узлах двунаправленной сети для сбора, обработки, анализа информации о функции потока в целях оценки потока в той части сети, которая непосредственно не наблюдается. Задача минимизации размера множества обозреваемых узлов (оптимальное решение) относится к классу NP-полных задач. Рассматривается задача поиска приемлемого числа обозреваемых узлов (субоптимальное решение).*

## ВВЕДЕНИЕ

Рассматривается актуальная задача расположения специальных программируемых устройств (сенсоров) в узлах двунаправленной сети для сбора информации о функции потока с целью оценки однородного потока в сети. Задача минимизации размера множества обозреваемых узлов, в которых локализованы сенсоры [1,2], потребует огромных вычислительных затрат, поскольку эта задача принадлежит классу NP-полных задач дискретной математики [3]. Для установления полной наблюдаемости сети относительно большой размерности осуществляется поиск приемлемого числа узлов (субоптимальное решение). Работа посвящена разработке стратегий идентификации расположения специальных программируемых устройств (сенсоров) в узлах сети для сбора, обработки, анализа информации о функции потока в целях оценки дуговых потоков в той части сети, которая непосредственно не наблюдается. В качестве модели потоковой сети с локализацией специальных программируемых устройств (сенсоров) в узлах двунаправленной сети используется разреженная недоопределенная система линейных алгебраических уравнений. Сформируем разреженную систему специального вида на основе априорной информации от сенсоров, установленных в обозреваемых узлах сети [4].

### I. МОДЕЛЬ ДВУНАПРАВЛЕННОЙ СЕТИ С ЛОКАЛИЗАЦИЕЙ СЕНСОРОВ В УЗЛАХ

Для каждого узла  $i \in I$  конечного, связанного, ориентированного двунаправленного графа  $G = (I, U)$  выполняются условия сохранения потока

$$\sum_{j \in I_i^+(U)} x_{i,j} - \sum_{j \in I_i^-(U)} x_{j,i} = \begin{cases} x_i, & i \in I^*, \\ 0, & i \in I \setminus I^*, \end{cases} \quad (1)$$

где  $x = (x_{i,j}, (i,j) \in U; x_i, i \in I^*)$  – вектор неизвестных дуговых и внешних потоков.

Для каждой дуги  $(i,j) \in U$  известно численное значение  $p_{i,j} \in (0, 1]$ , которое является долей

общего потока, исходящего из узла  $i \in I$ :

$$x_{i,j} = p_{i,j} \sum_{j \in I_i^+(U)} x_{i,j}, \quad \sum_{j \in I_i^+(U)} p_{i,j} = 1. \quad (2)$$

Узлы  $i \in M$  сети  $G$ , в которых размещены специальные программируемые устройства (сенсоры) – обозреваемые узлы,  $M \subseteq I$ . Дуга  $(i,j)$  – каноническая дуга узла  $i$ , если  $p_{i,j} \neq 0$ ,  $j \in I_i^+(U)$ ,  $i \in I$ . Для двунаправленного графа  $G$  при условии  $p_{i,j} \in (0, 1]$ ,  $(i,j) \in U$  существует каноническая дуга  $(i,k)$ ,  $k \in I_i^+(U)$  с ненулевым дуговым потоком  $x_{i,k}$  для каждого узла  $i \in I$ .

В результате размещения сенсоров в обозреваемых узлах  $M \subset I$  графа  $G$  известна следующая информация о функции потока:

- численные значения  $f_{i,j}$ ,  $f_{j,i}$  дуговых потоков соответственно для исходящих и входящих дуг каждого обозреваемого узла  $i \in M$ :

$$\begin{aligned} x_{i,j} &= f_{i,j}, \quad j \in I_i^+(U), \\ x_{j,i} &= f_{j,i}, \quad j \in I_i^-(U), \quad i \in M, \end{aligned} \quad (3)$$

- численные значения внешнего потока  $f_i$  в узлах  $i \in M \cap I^* \neq \emptyset$ :

$$x_i = f_i, \quad i \in M \cap I^*. \quad (4)$$

- численные значения дуговых потоков, полученные посредством известной информации, полученной от сенсоров:

$$\begin{aligned} x_{i,j} &= \beta_{i,j} f_{i,k_i}, \quad j \in I_i^+(U) \setminus \{k_i\}, \\ \beta_{i,j} &= \frac{p_{i,j}}{p_{i,k_i}}, \quad |I_i^+(U)| > 1; \\ \beta_{i,j} &= p_{i,k_i} = 1, \quad |I_i^+(U)| = 1. \end{aligned} \quad (5)$$

### II. НЕНАБЛЮДАЕМАЯ ЧАСТЬ СЕТИ

Построим граф  $\bar{G} = (\bar{I}, \bar{U})$  – ненаблюдаемую часть графа  $G$ . Удалим из графа  $G$  дуги и узлы, для которых известны численные значения (3) – (5). Система для вычисления неизвестных дуговых потоков графа  $\bar{G} = (\bar{I}, \bar{U})$  примет вид:

$$\sum_{j \in I_i^+(\bar{U})} x_{i,j} - \sum_{j \in I_i^-(\bar{U})} x_{j,i} = \begin{cases} x_i + b_i, & i \in \bar{I}^*, \\ b_i, & i \in \bar{I} \setminus \bar{I}^*, \end{cases} \quad (6)$$

$$x_{i,j} = \beta_{i,j} x_{i,k_i}, j \in I_i^+(\bar{U}) \setminus \{k\},$$

$$\beta_{i,j} = \frac{p_{i,j}}{p_{i,k_i}}, |I_i^+(\bar{U})| > 1;$$

где  $b_i, i \in \bar{I}$  – константы, полученные на основании априорной информации (3) – (5).

Граф  $\bar{G} = (\bar{I}, \bar{U})$  – ненаблюдаемая часть графа  $G = (I, U)$  – может быть несвязным. Некоторые компоненты связности графа  $\bar{G}$  могут не содержать узлов из множества  $\bar{I}^*$  с ненулевым внешним потоком. В этом случае базисный граф является **остовным деревом**. Для других компонент связности выполняется условие  $\bar{I}^* \neq \emptyset$  и базисный граф – **лес деревьев**, со специальными свойствами.

### III. УСЛОВИЯ ПОЛНОЙ НАБЛЮДАЕМОСТИ СЕТИ

Разреженная система (6) может быть одного из следующих типов:

- недоопределенная
- переопределенная
- имеет единственное решение

Основные вычислительные затраты по определению типа системы (6) (является недоопределенной, переопределенной или имеет единственное решение) относятся к вычислению ранга матрицы системы (6).

**Теорема 1.** Если для множества обозреваемых узлов  $M$  ранг матрицы системы (6) равен числу ее неизвестных, то система имеет единственное решение.

**Теорема 2.** Пусть  $A$  – матрица инцидентности связного ориентированного графа  $G$ . Строки усеченной матрицы  $\tilde{A}$ , образованной удалением любой строки матрицы  $A$ , линейно независимы.

**Теорема 3.** Пусть  $G$  – связный ориентированный двунаправленный граф с функцией потока (1), содержащий  $k = |I^*|$  узлов с внешним потоком  $x_i, k \neq 0$ . Известны коэффициенты  $p_{i,j}, (i, j) \in U$  разбиения потока. Для определения численных значений дуговых потоков всего графа достаточно разместить  $k = |I^*|$  сенсоров в узлах множества  $I^*$ . [4]

Верхняя граница  $\bar{h}$  интервала  $[\underline{h}, \bar{h}]$ , где  $\underline{h} = 1, \bar{h} = |I|$ , изменения количества обозреваемых узлов графа  $G = (I, U)$  субоптимального решения уменьшена до значения  $\bar{h} = |I^*|, I^* \neq \emptyset$ . Установка специальных программируемых устройств (сенсоров) в узлы множества  $I^*$  гарантирует полную наблюдаемость сети  $G$ .

Эффективный метод декомпозиции однородного потока разработан в [5, 6].

Подмножество  $M \subseteq I^*$  обозреваемых узлов сети  $G$  –  $t$ -оптимальное решение задачи идентификации сенсорной конфигурации,  $|M| \in [\underline{h}, \bar{h}]$ ,  $\underline{h} = 1, \bar{h} = |I^*|$ , если выполняются условия:  $|x_i| \geq t, i \in I^*, x_i = 0, i \in I \setminus I^*, t \in [\underline{t}, \bar{t}]$  и система (6) имеет единственное решение. Для заданного внешнего потока  $t, |x_i| \geq t, i \in I^*$  найти подмножество  $M \subseteq I^*$  обозреваемых узлов графа  $G, |M| \in [1, |I^*|], |M| \leq |I^*|$ , такое, что система (6) для графа  $\bar{G} = (\bar{I}, \bar{U})$  – ненаблюдаемой части графа  $G$  имеет единственное решение (полная наблюдаемость сети  $G$ ).

Для дальнейшего уменьшения числа  $|I^*|$  обозреваемых узлов и построения сенсорной конфигурации множества узлов  $M \subseteq I^*$   $t$ -оптимального решения для заданного порога интенсивности  $t$  применяются стратегии случайного поиска числа  $|M|$ , принадлежащего интервалу  $[\underline{h}, \bar{h}]$ , где  $\underline{h} = 1, \bar{h} = |I^*|$ .

### ЗАКЛЮЧЕНИЕ

Исследованы свойства систем с различными типами разреженности, которые возникают при построении оптимальных и субоптимальных ( $t$ -оптимальных) решений задачи оценки однородного потока в сети. В синтезе с современными инновационными технологиями разреженного матричного анализа, теории графов, теоретической информатики созданы эффективные численные методы декомпозиции. Получены условия полной наблюдаемости однородного потока для случая локализации специальных программируемых устройств (сенсоров) в узлах двунаправленной сети.

1. Gentile M. Locating active sensors on traffic networks / Gentile M., Mirchandani P. // Annals of Operation Research. – 2006. – Vol. 144, № 1. – P. 201–234.
2. Bianco L. Combinatorial aspects of the Sensor Location Problem / Bianco L., Confessore G., Gentili M. // Annals of Operation Research. – 2006. – Vol. 144, № 1. – P. 201–234.
3. Bianco L. A network based model for traffic sensor location with implications on O/D matrix estimates / Bianco L., Confessore G., Reverbery P. // Transportation Science. – 2001. – Vol. 35, № 1. – P. 50–60.
4. Пилипчук Л. А. Идентификация сенсорной конфигурации и управление потоками / Пилипчук Л. А., Пилипчук А. С., Полячок Е. Н., Фаразей А. И. // Журнал Белорусского государственного университета. Математика. Информатика. – 2018. – № 2. – С. 67–76.
5. Pilipchuk L. A. Algorithms of Solving Large Sparse Underdetermined Linear Systems with Embedded Network Structure / Pilipchuk L. A., Malakhouskaya Y. V., Kincaid D. R., Lai M. // East-West J. of Mathematics. – 2002. – Vol. 4, № 2. – P. 191–201.
6. Пилипчук, Л. А. Разреженные недоопределенные системы линейных алгебраических уравнений / Л. А. Пилипчук. – Минск: БГУ, 2012. – 260 с.

# ЧИСЛЕННОЕ ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ МЕТОДОВ РЕШЕНИЯ ЗАДАЧ О КРАТЧАЙШИХ ПУТЯХ В ОРИЕНТИРОВАННЫХ ГРАФАХ

Полячок Е. Н.

Кафедра компьютерных технологий и систем, УО «Белорусский государственный университет»

Минск, Республика Беларусь

E-mail: arszp10@gmail.com

*Рассматривается проблема разработки алгоритмических, структурных и технологических решений задачи о кратчайших путях из заданного узла в достижимые узлы орграфа и исследование особенностей численной реализации методов. Алгоритмы Дейкстры и Беллмана-Форда основаны на применении принципа динамического программирования. Для базисного метода построены структурные и алгоритмические решения задачи о кратчайших путях с использованием двух технологий хранения стоимостей дуг графа.*

## ВВЕДЕНИЕ

Актуальным направлением научных и прикладных исследований является развитие алгоритмической теории графов для решения задач о кратчайших путях в орграфах. Исследование и создание численных методов и их реализаций базируется на двух подходах: 1) применение принципа динамического программирования и 2) использование корневых деревьев для представления и преобразования базисного множества дуг. В работе рассматриваются алгоритмические, структурные и технологические решения задач о кратчайших путях в ориентированных графах и численное исследование их эффективности.

### I. ИНДЕКСНЫЕ И БАЗИСНЫЕ МЕТОДЫ

В итерационных процессах индексных алгоритмов (Дейкстры, Беллмана-Форда) построение кратчайшего пути из начального узла  $s$  в достижимый узел  $j$  графа  $G = (V, E)$  основано на применении принципа динамического программирования. На каждой итерации выполняется решение уравнения Беллмана:

$$B_j = \min_{i \in V_j^-(E)} \{B_i + c_{i,j}\}, j \neq s, j \in V, \quad (1)$$

с начальным условием  $B_s = 0$ , где  $B_j$  – длина кратчайшего пути из начального узла  $s$  в узел  $j$  графа,  $V_j^-(E)$  – множество узлов графа, из которых исходят дуги, входящие в узел  $j$ .

Опорные (базисные) методы построения кратчайших путей основаны на исследовании математической модели [1], структурных свойств базисов, представленных с помощью корневых деревьев, технологии преобразования допустимых решений на итерациях поиска кратчайших путей из начального узла  $s$  в достижимые узлы графа  $G$ .

### II. СТРУКТУРНЫЕ СВОЙСТВА И ТЕХНОЛОГИИ

## РЕАЛИЗАЦИИ

Существует несколько способов поиска дуги, на которой нарушаются условия оптимальности [1]: а) просмотр дуг в произвольном порядке, пока не встретится дуга, свидетельствующая о нарушении условий оптимальности текущего опорного потока; б) просмотр дуг пока не встретится дуга, свидетельствующая о *минимальной оценке* нарушения условий оптимальности; в) хранение дуг в кольцевом списке. Каждый раз, когда требуется найти дугу, на которой нарушаются условия оптимальности, ее поиск начинается с дуги, следующей за той, на которой такой поиск остановился на предыдущем шаге алгоритма. Если на очередном шаге алгоритма при поиске было пройдено  $m = |E|$  дуг, то алгоритм завершается, так как текущий опорный поток является оптимальным. Этот способ поиска дуги графа с нарушением условий оптимальности дает лучшие временные характеристики.

В памяти компьютера стоимости дуг могут храниться в списке инцидентности графа  $G$  (узел, дуга, стоимость дуги). Если из узла исходит  $k$  дуг, то сложность алгоритма поиска стоимости дуги составляет  $O(k)$  операций в худшем случае [2]. Можно воспользоваться хеш-таблицей, где ключом является дуга, а значением ее стоимость. Сложность алгоритма поиска значения в хеш-таблице по ключу составляет  $O(1)$  и  $O(n)$  операций в среднем и худшем случаях соответственно.

В [3] получен результат, позволяющий восстановить потоки по дугам оптимального корневого дерева.

### III. ГЕНЕРАЦИЯ ГРАФОВ

Для генерации ациклических ориентированных графов, не содержащих ориентированных циклов, можно воспользоваться заранее определенным топологическим порядком графа, например, топологическим порядком  $1, 2, \dots, n$ ,



где  $n$  – количество узлов генерируемого графа. При необходимости генерации графа, в котором не все узлы достижимы из начального узла, на вход алгоритма также должно поступать количество дуг генерируемого графа. Пока количество дуг при генерации не достигнет желаемого значения, генерируются два случайных целых числа из интервале  $[1, n]$  – узлы и стоимость генерируемой дуги. В множество дуг графа добавляется новая дуга в зависимости от порядка следования этих двух узлов в топологическом порядке, если такая дуга не принадлежит графу. Пусть сгенерированы два узла  $u$  и  $v$ . Если узел  $v$  следует после узла  $u$  в топологическом порядке генерируемого графа, то в граф можно добавить дугу  $(u, v)$ , иначе  $(v, u)$ . Начальным узлом для поиска кратчайших путей в полученном графе можно положить любой из узлов.

Таким образом, максимально возможное количество сгенерированных дуг равно  $(n-1)/2$ , так как первый узел топологического порядка можно соединить не более чем с  $n-1$  узлом, второй узел – с  $n-2$  узлами, и так далее. При генерации графа, в котором все узлы достижимы из начального узла на каждом шаге после генерации очередной дуги и ее стоимости нужно проверять существование узлов в генерируемом графе, не достижимых из начального узла. Если существуют, то генерацию дуг нужно продолжить.

#### IV. ВРЕМЕННЫЕ ХАРАКТЕРИСТИКИ

Реализованы два варианта базисного алгоритма, различающиеся способом хранения стоимостей дуг графа: хранение стоимостей дуг внутри списков инцидентности (алгоритм A1) и хранение стоимостей дуг в хеш-таблице (алгоритм A2). Временные характеристики в секундах для графов, в которых достижимы все узлы из начального узла представлены в таблице 1.

Таблица 1 – Временные характеристики алгоритмов A1 и A2

| $ V $ | $ E $   | $t, A1$ | $t, A2$ | $t_1/t_2$ |
|-------|---------|---------|---------|-----------|
| 100   | 4 950   | 0.1082  | 0.0325  | 3.33      |
| 200   | 19 900  | 0.4587  | 0.1355  | 3.39      |
| 300   | 44 850  | 1.1780  | 0.2489  | 4.73      |
| 400   | 79 800  | 2.2919  | 0.4512  | 5.08      |
| 500   | 124 750 | 4.3207  | 0.6661  | 6.49      |
| 750   | 280 785 | 13.6831 | 1.4021  | 9.76      |
| 1 000 | 499 500 | 31.7400 | 2.5531  | 12.43     |

Исходя из данных таблицы явно прослеживается тот факт, что время работы базисного алгоритма с хранением стоимостей дуг в списках инцидентности при увеличении размерностей графа растет гораздо быстрее, чем при использовании хеш-таблицы. Следовательно, в дальнейших экспериментах, для хранения стоимостей дуг используется хеш-таблица.

В работе реализованы два алгоритма поиска кратчайших путей в графе: алгоритм Дейкстры и алгоритм Беллмана-Форда. Сравнение вре-

менных характеристик этих алгоритмов приведены в таблице 2.

Таблица 2 – Временные характеристики алгоритмов Беллмана-Форда (B1), базисного метода (B2) и алгоритма Дейкстры (B3).

| $ V $ | $ E $   | Время работы алгоритмов, с |        |        |
|-------|---------|----------------------------|--------|--------|
|       |         | B1                         | B2     | B3     |
| 100   | 4 950   | 0.0671                     | 0.0325 | 0.0008 |
| 200   | 19 900  | 0.1942                     | 0.1355 | 0.0035 |
| 300   | 44 850  | 0.2697                     | 0.2489 | 0.0078 |
| 400   | 79 800  | 0.5222                     | 0.4512 | 0.0148 |
| 500   | 124 750 | 1.3862                     | 0.6661 | 0.0232 |
| 750   | 280 875 | 4.2057                     | 1.4021 | 0.0509 |
| 1 000 | 499 500 | 7.6665                     | 2.5531 | 0.0932 |

В экспериментах использовалась реализация алгоритма Дейкстры с использованием кучи Фибоначчи. Однако для алгоритма Дейкстры стоимости дуг должны быть положительными, для алгоритмов Беллмана-Форда и базисного метода стоимости могут быть любого знака.

В таблице 3 приведено отношение времен работы алгоритма Беллмана-Форда к базисному методу.

Таблица 3 – Отношение времени  $t_1$  работы алгоритма Беллмана-Форда (B1) и  $t_2$  – базисного алгоритма (B2)

| $ V $ | $ E $   | $t_1, c$ | $t_2, c$ | $t_1/t_2$ |
|-------|---------|----------|----------|-----------|
| 100   | 4 950   | 0.0671   | 0.0325   | 2.06      |
| 200   | 19 900  | 0.1942   | 0.1355   | 1.43      |
| 300   | 44 850  | 0.2697   | 0.2489   | 1.08      |
| 400   | 79 800  | 0.5222   | 0.4512   | 1.16      |
| 500   | 124 750 | 1.3862   | 0.6661   | 2.08      |
| 750   | 280 875 | 4.2057   | 1.4021   | 3.00      |
| 1 000 | 499 500 | 7.6665   | 2.5531   | 3.00      |

#### ЗАКЛЮЧЕНИЕ

Разработан генератор связанных ориентированных ациклических графов. Реализован базисный метод поиска кратчайших путей из начального узла в достижимые узлы с использованием двух технологий хранения стоимости дуг графа и проведено сравнение его быстродействия. Выполнено численное исследование эффективности трех методов решения задач о кратчайших путях в ориентированных ациклических графах.

1. Пилипчук, Л.А. Оптимальные пути: алгоритмические, структурные и технологические решения / Л.А. Пилипчук, А.С. Пилипчук, Е.Н. Полячок // Веснік Гродзенскага дзяржаўнага ўніверсітэта імя Янкі Купалы. Серыя 2. Матэматыка. Фізіка. Інфарматыка, вылічальная тэхніка і кіраванне. – Т. 10. №3. – 2020. – С. 143–151.
2. Йенсен, П. Поточное программирование. Пер. с англ. / П. Йенсен, Д. Барнес. – М.: Радио и связь, 1984. – 392 с.
3. Пилипчук Л.А. О численных методах и технологиях построения кратчайших путей / А.С. Пилипчук, Е.Н. Полячок, А.Э. Лобко, Д.А. Шкурский // Материалы международной научной конференции «Информационные технологии и системы 2020 (ITS 2020) = Information Technologies and Systems 2020 (ITS 2020)». – Республика Беларусь. Минск, 18 ноября 2020 г. – Минск: БГУИР. – 2020. – С. 136–137.

# РЕАЛИЗАЦИЯ АЛГОРИТМОВ ТРИАНГУЛЯЦИИ ДЛЯ ТВЕРДОТЕЛЬНОГО МОДЕЛИРОВАНИЯ

Прорашнев В. Л.

Факультет информационных технологий и управления  
Белорусского государственного университета информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: prorashnev.vsevolod@mail.ru

*В работе рассматриваются оптимизации алгоритма триангуляции для пакетов 3д-моделирования.*

## РАССМОТРЕНИЕ ВАРИАНТОВ ОПТИМИЗАЦИИ АЛГОРИТМА ТРИАНГУЛЯЦИИ

Процедура триангуляции в вычислительной геометрии - это разложение многоугольной фигуры  $P$  на некоторое множество треугольников, т.е. нахождение множества треугольников с попарно непересекающимися внутренними областями, объединением которых является  $P$ . Триангулировать любой выпуклый многоугольник за линейное время в верную триангуляцию триангуляционно, добавляя диагонали от одной вершины ко всем остальным вершинам. Общее количество способов триангулировать выпуклый  $n$ -угольник непересекающимися диагоналями - это  $(n-2)$ -е число Каталана, которое можно вычислить по формуле (1), открытой Леонардом Эйлером.

$$C_{n-2} = \frac{n(n+1) \dots (2n-4)}{(n-2)!}. \quad (1)$$

Согласно данной формуле на примере выпуклого семиугольника (7-стороннего выпуклого многоугольника) имеем 42 возможных триангуляции (рисунок 1). Это количество задается 5-м каталанским числом.

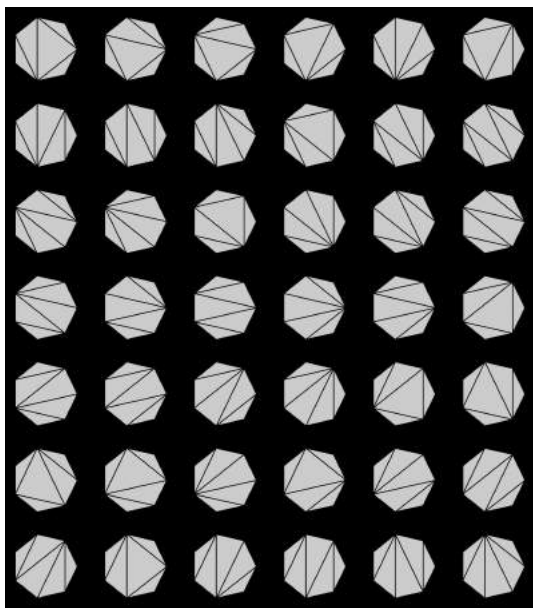


Рис. 1 – 42 варианта триангуляции семиугольника

Однако, верная триангуляция перестает быть актуальной при рассмотрении более слож-

ных многоугольников, не удовлетворяющих требованию выпуклости. Для такого случая существуют методы «обрезания ушей» и «монотонной триангуляции». [1]

Метод обрезания ушей основывается на том, что любой простой многоугольник с минимум 4 вершинами без отверстий имеет как минимум два «уха» – треугольники, две стороны которых являются краями многоугольника, а третья сторона находится полностью внутри него. Алгоритм состоит в том, чтобы найти такое ухо, удалить его из многоугольника (в результате чего получается новый многоугольник, который все еще удовлетворяет условиям) и повторять, пока не останется только один треугольник. Этот алгоритм прост в реализации, но работает медленнее, чем некоторые другие алгоритмы, и он работает только на многоугольниках без отверстий. Эффективный алгоритм обрезания ушей был найден Хоссамом Эль Гинди, Хейзел Эверетт и Годфридом Туссенем.

Монотонная триангуляция многоугольников заключается в разбиении многоугольника на монотонные многоугольники. Простой многоугольник является монотонным относительно прямой  $L$ , если любая прямая, ортогональная к  $L$ , пересекает многоугольник не более двух раз. Монотонный многоугольник может быть разбит на две монотонные цепи. Многоугольник, монотонный относительно оси  $y$ , называется  $y$ -монотонным. Предполагая, что данный многоугольник является  $y$ -монотонным, алгоритм начинается с прохождения по одной цепи многоугольника сверху вниз, добавляя диагонали при каждой возможности. Данный алгоритм может быть эффективно применен к любому монотонному многоугольнику. Если многоугольник не является монотонным, его можно разбить на монотонные подполигоны, используя подход с использованием линий развертки. Алгоритм не требует, чтобы многоугольник был простым, поэтому его можно применять к многоугольникам с отверстиями.

В рамках компьютерной графики триангуляция является неотъемлемым этапом подготовки полигональных объектов к работе, поскольку 3д пакеты, использующие полигональные модели, в конечном итоге предназначены для работы с треугольниками. При попытке использо-

вания в работе многоугольников с большим количеством вершин велика вероятность неверного отображения визуализации объекта. Исключением для этой процедуры могут являться четырехугольные многоугольники за счет меньшей склонности к созданию ошибок отображения в связи с малым количеством вершин и вариантов триангуляции. Однако, в трехмерном пространстве триангуляция многоугольника не дает гарантий правильного его отображения. Наиболее актуальна данная проблематика для сферы анимации, когда при деформации триангулированного многоугольника его вершины перемещаются неравномерно, тем самым создавая новые, более сложные формы из образованных во время триангуляции диагоналей. Анализируя условия возникновения некорректного отображения объекта при деформации триангулированного многоугольника, можно прийти к выводу, одной из основных причин является сонаправленность ребер на внутренней площади многоугольника, что обусловлено недопустимостью пересечения внутренних граней по определению триангуляции. Таким образом, для снижения вероятности ошибок визуализации триангулированных многоугольников требуется модификация алгоритма триангуляции, не ограниченная требованиями о непересечении внутренних граней. Логичным шагом для получения данной модификации будет добавление дополнительных вершин в центр периметра многоугольника. Поскольку в общем случае неизвестно, достаточно ли будет данной операции для избежания ошибок визуализации, следует добавить итеративность в этап добавления вершин. На первой итерации одна дополнительная вершина помещается в центр массы многоугольника. Алгоритм для нахождения центра массы объекта в двух и трех измерениях уже реализован для всех пакетов трёхмерного проектирования, работающих с полигональными объектами, и подходит для применения в рамках рассматриваемой модификации алгоритма триангуляции. На второй итерации рассматриваются отсекаемые прилегающими к дополнительной вершине ребрами участки многоугольника и для участка с наибольшей площадью находится центр массы и создается новая дополнительная вершина. Последующие итерации повторяют процедуру, рассматривая отсекаемые набором ребер, прилегающих к дополнительным вершинам, участки многоугольника. На рисунках 2-3 представлено сравнение результата триангуляции с результатом двух итераций модифицированного алгоритма на примере случайно сгенерированного  $n$ -угольника, где  $n$  равно 32.

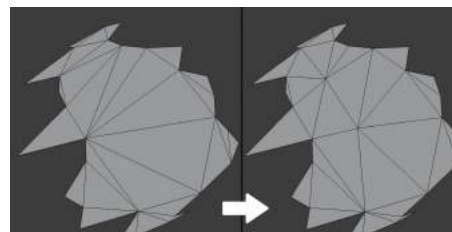


Рис. 2 – Пример оптимизации алгоритма триангуляции

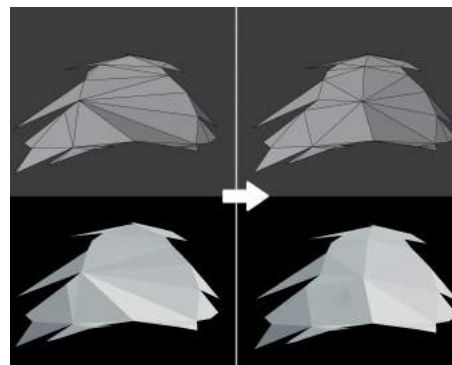


Рис. 3 – Пример визуализации многоугольника до модификации алгоритма триангуляции и после

В рассмотренном примере два одинаковых многоугольника подвергались одинаковой деформации, - сгиб по оси  $y$ . Добавление всего двух новых вершин помогло избежать потери формы и, как следствие, ошибок визуализации после триангуляции. Однако, в примере рассматривалась деформация двухмерной фигуры, поскольку все вершины исходного многогранника находились в одной плоскости. Рассмотрим пример с трехмерной фигурой, часть которой заменим сложным  $n$ -угольником.

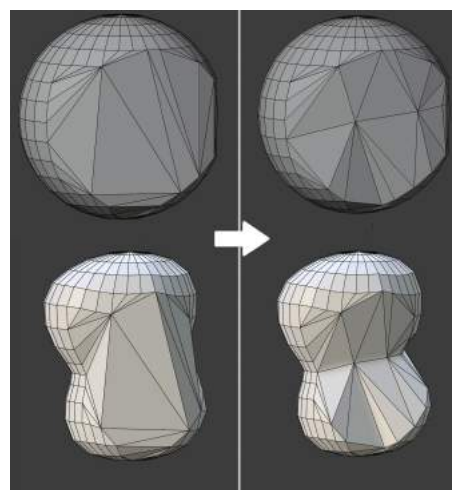


Рис. 4 – Пример визуализации многоугольника до модификации алгоритма триангуляции и после

## I. СПИСОК ЛИТЕРАТУРЫ

1. Proceedings of the 2012 International Conference on Information Technology and Software Engineering / Ed. J. Wei Lu, Guoqiang Cai, Weibin Liu // Springer-Verlag Berlin Heidelberg. – 2013. – P.980-984.

# МЕТОДЫ И СРЕДСТВА СИСТЕМЫ ГЕОБАЗАДАННЫХ ДЛЯ АДАПТАЦИИ КОМПЬЮТЕРНЫХ МОДЕЛЕЙ. ИНСТРУМЕНТЫ КЛАСТЕРИЗАЦИИ

Таранчук В. Б.

Кафедра компьютерных технологий и систем, Факультет прикладной математики и информатики,  
Белорусский государственный университет  
Минск, Республика Беларусь  
E-mail: taranchuk@bsu.by

*Обсуждаются методические и технические вопросы развития программной системы ГеоБазаДанных (ГБД). Отмечены новые функциональные возможности, обеспеченные включением в ГБД исполняемых модулей интеллектуального анализа данных системы компьютерной алгебры Wolfram Mathematica. В частности, подготовлены и предполагается обсудить на представительных наборах геоданных варианты выбора наилучших алгоритмов кластеризации.*

## ВВЕДЕНИЕ

Решением задачи кластерного анализа (сегментации) являются разбиения, удовлетворяющие принимаемому критерию. Критерий обычно представляет собой формализованный функционально набор правил для определения уровней различий при разбиениях и группировках (целевая функция). Кластерный анализ широко применяется во многих областях, в частности, в компьютерных системах при распознавании образов, анализе изображений, поиске информации, сжатии данных, в компьютерной графике, биоинформатике, машинном обучении. При интеллектуальном анализе данных сегментация может использоваться как самостоятельный инструмент для принятия решения о распределении данных, для контроля характеристик и последующего анализа наборов данных, конфигураций и содержания конкретных кластеров. В качестве альтернативы, кластерный анализ может служить этапом предварительной обработки для других алгоритмов. Также сегментация используется для обнаружения нетипичных объектов – выбросов (значения, которые находятся «далеко» от любого кластера), это – обнаружение новизны, такие объекты могут быть более интересными, чем включенные в кластеры.

Важное достоинство кластерного анализа в том, что при его выполнении можно производить разбиение объектов не только по одному параметру, а по набору признаков. Кроме того, кластерный анализ в отличие от большинства математико-статистических методов не накладывает никаких ограничений на вид рассматриваемых исходных данных.

### I. КРАТКО О ГЕОБАЗАДАННЫХ

ГеоБазаДанных – комплекс интеллектуальных компьютерных подсистем, математического, алгоритмического и программного обеспечения наполнения, сопровождения и визуализации баз данных, входных данных для имитационных и математических моделей, средств проведения

вычислительных экспериментов, алгоритмических и программных средств создания постоянно действующих компьютерных моделей [1]. Средствами ГБД можно формировать и визуализировать цифровые описания пространственных распределений данных об источниках загрязнения, о геологическом строении изучаемых объектов; графически иллюстрировать решения задач, описывающих динамические процессы многофазной фильтрации, миграции флюидов, переноса тепла, влаги, минеральных водорастворимых соединений в толщах пород; конструировать и реализовывать интерактивные сценарии визуализации и обработки результатов вычислительных экспериментов. Подсистемы ГБД позволяют рассчитывать и выполнять в разных приближениях экспертные оценки локальных и интегральных характеристик экосистем, выполнять расчет распределений концентраций и массовых балансов загрязняющих веществ; создавать постоянно действующие модели объектов нефтедобычи; формировать и выводить на твердые копии тематические карты.

### II. ПОДГОТОВКА ИСХОДНЫХ ДАННЫХ

Примеры, включаемые для обсуждения в докладе, рассчитаны с данными [2]. Напомним, что моделируемая поверхность (рис. 1 [2]) имеет полное математическое описание. Данные для демонстраций методов и алгоритмов интеллектуального анализа получены имитацией замеров, соответствующий набор данных – пункты замеров уровня восстанавливаемой поверхности, представляющие (по факту) рассеянное множество точек, интерпретируются, как данные на профилях наблюдений. Соответствующая схема их размещения показана на рис. 2, 3 [2].

### III. ИНСТРУМЕНТАРИЙ, ПРИМЕРЫ КЛАСТЕРНОГО АНАЛИЗА ГЕОДААННЫХ

**Эффекты числа кластеров.** Одной из важнейших проблем сегментации является определение количества кластеров. Определение ко-

личества кластеров – это одна из важнейших проблем сегментации. В более широком смысле – это проблема инициализации алгоритма: выбора оптимальных значений управляющих параметров, используемых оценочных функций, метрики, условий останковки и т.п. По результатам выполненной первой серии расчетов в докладе будут приведены иллюстрации, рассчитанные с установками по умолчанию, используя функцию Wolfram Mathematica FindClusters. Сопоставление вариантов дает основание утверждать, что необходимы дополнительные действия по выбору метода, метрики и других параметров алгоритмов кластеризации.

**Эффекты принятого метода кластеризации.** Во второй серии будут представлены результаты, которые иллюстрируют особенности наиболее часто используемых алгоритмов кластеризации. Кластеризация в примерах этой серии рассматривалась только для пар координат, т.е. учитывалось относительное положение точек рассеянного множества, причем, в программном модуле использована функция FindClusters с разными критериями CriterionFunction, норма в примерах серии 2 вычислялась по метрике DistanceFunction EuclideanDistance. Вообще говоря, включенное в ГеоБазаДанных из системы Wolfram Mathematica соответствующее программное приложение допускает варианты метода кластеризации (CriterionFunction): Automatic, find clustering hierarchically, find clustering by local optimization, density-based spatial clustering of applications with noise, variational Gaussian mixture algorithm, Patrick clustering algorithm, k-means clustering algorithm, partitioning around medoids, mean-shift clustering algorithm, displace examples toward high-density region, minimum spanning tree-based clustering algorithm, spectral clustering algorithm.

**Влияние метрики.** В рассмотренных примерах серий 1 и 2, а также в третьей серии результаты сходство или различие между классифицируемыми объектами устанавливается в зависимости от метрического расстояния между ними. Вопросы измерения близости объектов приходится решать при любых трактовках кластеров и различных методах классификации, причем, имеют место неоднозначность выбора способа нормировки и определения расстояния между объектами. Влияние метрики (DistanceFunction) будет проиллюстрировано схемами серии 3. Приведенные в этой серии результаты получены средствами включенного в ГеоБазаДанных из системы Wolfram Mathematica соответствующего программного приложения, которое допускает разные варианты задания DistanceFunction (Possible settings for Method). В системе Wolfram Mathematica различные меры расстояния или сходства удобны для различных типов анализа. Язык Wolfram

предоставляет встроенные функции для многих стандартных измерений расстояния, а также возможность давать символическое определение произвольной меры. В частности, для анализа цифровых данных доступны следующие варианты метрик: Euclidean Distance, Squared Euclidean Distance, Normalized Squared Euclidean Distance, Manhattan Distance, Chessboard Distance, Bray Curtis Distance, Canberra Distance, Cosine Distance, Correlation Distance, Binary Distance, Warping Distance, Canonical Warping Distance.

Резюмируя серии 1–3 можно отметить, что для рассматриваемой конфигурации точек с данными, учитывая цифровое поле оригинала, однозначно назвать какой-то из рассчитанных вариантов предпочтительным трудно.

#### IV. ВЛИЯНИЕ УЧЕТА ЗНАЧЕНИЙ В ТОЧКАХ

В рассмотренных примерах серий 1–3, в результатах сходство или различие между классифицируемыми объектами устанавливается в зависимости от метрического расстояния между ними. Отдельно выполнен анализ, получены варианты классификации с использованием функции Wolfram Mathematica Cluster Classify, которая позволяет выполнять кластеризацию не только, принимая во внимание координаты точек рассеянного множества, но и значения в них. Другими словами – в рассмотренных в четвертой серии результатах в алгоритмах учитываются не пары координат, а тройки – координаты и значение в каждой точке.

Из полученных результатов следует, что для рассматриваемого набора данных учет значений в точках дополнительного явно положительного эффекта в реализации кластеризации не дает. Но подобные результаты полезны и важны, так как из сопоставления ясно, где необходимы дополнительные исходные данные.

#### ЗАКЛЮЧЕНИЕ

Рассматриваются вопросы инструментального наполнения и использования интерактивной компьютерной системы ГеоБазаДанных. Представлены и обсуждаются результаты кластеризации представительного набора данных типичной цифровой модели пространственного объекта.

#### V. СПИСОК ЛИТЕРАТУРЫ

1. Таранчук, В. Б. Компьютерные модели подземной гидродинамики / В. Б. Таранчук // Минск : БГУ, 2020. – 235 с.
2. Таранчук, В. Б. Методы и средства системы ГеоБазаДанных для адаптации компьютерных моделей. Примеры адаптации / В. Б. Таранчук // Информационные технологии и системы 2020 (ИТС 2020) : материалы междунар. науч. конф. (Республика Беларусь, Минск, 18 ноября 2020 года) = Information Technologies and Systems 2020 (ITS 2020). –Minsk, – 2020. – С. 15–17.

# ОБ ОДНОМ СВОЙСТВЕ РЕШЕНИЙ СЕМЕЙСТВА ТРЕХМЕРНЫХ ПЯТИЭЛЕМЕНТНЫХ ДИССИПАТИВНЫХ ДИНАМИЧЕСКИХ СИСТЕМ С ХАОТИЧЕСКИМ ПОВЕДЕНИЕМ

Цегельник В. В.

Кафедра высшей математики,

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: tsegvv@bsuir.by

*Исследован характер возможных подвижных особых точек решений семейства трехмерных пятиэлементных диссипативных динамических систем с хаотическим поведением. Показано, что ни одна из девяти систем данного семейства не является системой Пенлеве -типа.*

## ВВЕДЕНИЕ

Построение как можно более простой хаотической системы является интересной и значимой темой в изучении динамических систем, а также практических применений в безопасной связи и широкополосной генерации сигналов. Основываясь на классических работах [1, 2], Спротт [3, 4] построил с помощью компьютерного моделирования ряд простых автономных дифференциальных уравнений третьего порядка с несколькими квадратичными нелинейностями и показал, что они обладают хаотическим поведением. Работы Спротта ставят вопрос, насколько простой может быть трехмерная автономная система непрерывного времени, если она хаотична. Ответ на данный вопрос был частично дан в работах [5, 6]. В [5] было показано, что все диссипативные трехмерные автономные квадратичные системы с общим числом членов в правых частях равным 4, нехаотичны. Аналогичный результат для консервативных систем (за исключением одной) был получен в [6].

В работе [7] исследованы качественные свойства решений класса трехмерных пятиэлементных диссипативных динамических систем с одной квадратичной нелинейностью. Используя также систематический компьютерный поиск в [7] было установлено наличие хаоса (в частности, странных аттракторов) в системах дифференциальных уравнений

$$\dot{x} = y^2 - x + Az, \dot{y} = x, \dot{z} = y, \quad (1)$$

$$\dot{x} = yz - x + Ay, \dot{y} = z, \dot{z} = x, \quad (2)$$

$$\dot{x} = y^2 + Az, \dot{y} = x - y, \dot{z} = y, \quad (3)$$

$$\dot{x} = yz + Az, \dot{y} = x - y, \dot{z} = x, \quad (4)$$

$$\dot{x} = yz + Az, \dot{y} = x - y, \dot{z} = y, \quad (5)$$

$$\dot{x} = -x + y + A, \dot{y} = xz, \dot{z} = y, \quad (6)$$

$$\dot{x} = yz - x, \dot{y} = z + A, \dot{z} = x, \quad (7)$$

$$\dot{x} = -x + z, \dot{y} = x + A, \dot{z} = xy, \quad (8)$$

$$\dot{x} = -x + z, \dot{y} = z + A, \dot{z} = xy \quad (9)$$

при определенных значениях параметра  $A$ . Каждая из систем (1)–(9) является диссипативной.

## I. ПОСТАНОВКА ЗАДАЧИ

Целью работы является исследование характера возможных подвижных особых точек (т.е. точек, положение которых зависит от начальных условий) решений систем (1)–(9) с неизвестными функциями  $x, y, z$  в предположении, что независимая переменная  $t$  является комплексной.

Системы (уравнения), общие решения которых не имеют подвижных критических особых точек, называют системами (уравнениями) Пенлеве -типа или P -типа.

## II. АЛГОРИТМ

Для решения поставленной задачи использован тест Пенлеве [8], представляющий набор условий, необходимых для отсутствия у общего решения системы дифференциальных уравнений подвижных критических особых точек (свойство Пенлеве). Тест [8] применяется к системам дифференциальных уравнений, представленных в виде

$$x'_i \cdot Q_i(x_1, x_2, \dots, x_n, \tau) = P_i(x_1, x_2, \dots, x_n), \quad i = \overline{1, n}. \quad (10)$$

Здесь  $\tau = t - t_0$ ,  $t_0$  — произвольная точка области  $G \subset C$ , а  $P_i, Q_i$  — полиномы от  $x_1, x_2, \dots, x_n$  с аналитическими по  $\tau$  коэффициентами;  $' = \frac{d}{d\tau}$ . Тест представляет собой последовательное выполнение следующих шагов.

Первый шаг. Определяются наименьшие возможные степени переменной  $\tau$  в формальных разложениях  $i$ -х компонент решения системы уравнений (10) вида

$$x_i = \sum_{j=0}^{\infty} c_{ij} \tau^{j-p_i}, \quad i = \overline{1, n}. \quad (11)$$

Для определения пар  $(c_0, p)$ ,  $c_0 = (c_{10}, c_{20}, \dots, c_{n0})$ ,  $p = (p_1, p_2, \dots, p_n)$  в систему (10) подставляются выражения (11). Если не все  $c_{i0} = 0$  и полученные  $p_i \in \mathbb{N}$ , то первый шаг пройден и анализ системы может быть продолжен.

Второй шаг. Он связан с отысканием индексов Фукса для каждой пары  $(c_0, p)$ . Индексами Фукса или резонансами называются номера  $j_r$  коэффициентов  $c_{ij}$  в разложениях  $i$ -х компонент решения (11), для которых коэффициенты  $c_{ij_r}$  содержат произвольный параметр. Индексы Фукса определяются подстановкой выражений  $x_i \sim c_{i0}\tau^{-p_i} + \beta_i\tau^{r-p_i}$  ( $i = \overline{1, n}$ ) в ведущие члены уравнений системы (10). Приравнявая выражения, линейные относительно  $\beta_k$  ( $k = \overline{1, n}$ ), нулю, получаем систему линейных алгебраических уравнений. Определитель матрицы коэффициентов этой системы, приравненный к нулю, определяет уравнение резонансов  $Q(r) = 0$ .

В случае, когда для каждой пары  $(c_0, p)$  все корни уравнения резонансов являются целыми, а отличные от  $-1$  и возможно, от нуля — положительными числами, второй шаг формального теста считается пройденным.

Третий шаг. Если все коэффициенты в разложении (11) могут быть определены и число произвольных среди них коэффициентов равно  $n - 1$ , то считается, что система дифференциальных уравнений (10) проходит третий шаг, а вместе с ним проходит и весь формальный тест Пенлеве.

Для анализа решений систем (1)–(9) использован также подход, заключающийся в замене каждой из них эквивалентным уравнением третьего порядка и сравнением его с известными уравнениями, являющимися уравнениями Пенлеве -типа.

### III. ВЫВОДЫ

Справедливы следующие утверждения

Теорема 1. Системы (1), (3) эквивалентны уравнению

$$\ddot{y} + \ddot{y} - 2y\dot{y} - Ay = 0, \quad (12)$$

а системы (7), (8) — уравнению

$$\ddot{y} + \ddot{y} - y\dot{y} + Ay = 0. \quad (13)$$

Системы (2), (4), (5), (6), (9) эквивалентны соответственно уравнениям

$$\ddot{y} + \ddot{y} - y\dot{y} - Ay = 0, \quad (14)$$

$$z \ddot{z} - \dot{z}\dot{z} = z^2\dot{z} - z\dot{z} + Az^2, \quad (15)$$

$$\dot{z} + \dot{z} - z\dot{z} - Az = 0, \quad (16)$$

$$z \ddot{z} - \dot{z}\dot{z} = -z\dot{z} + z^2\dot{z} + Az^2, \quad (17)$$

$$y \ddot{y} - \dot{y}\dot{y} = -y\dot{y} + y^2\dot{y} - Ay^2. \quad (18)$$

Теорема 2 [9]. Ни одно из уравнений (12)–(14), (16) не является уравнением Пенлеве -типа.

Следствие 1. Ни одна из систем (1)–(3), (5), (7), (8) не является системой Р -типа.

Теорема 3. Система (4) не является системой Пенлеве -типа.

Справедливость данного утверждения следует из того, что система (4) не проходит третий шаг теста Пенлеве.

Следствие 2. Уравнения (15), (17), (18) не являются уравнениями Р -типа.

Поскольку уравнение (15) эквивалентно системе (4), а уравнение (18) получается из (15) заменой  $z \rightarrow y$ ,  $A \rightarrow -A$ , то на основании теоремы 3 они не являются уравнениями Пенлеве -типа.

Следствие 3. Системы (6), (9) также не являются системами Р -типа.

### IV. СПИСОК ЛИТЕРАТУРЫ

1. Lorenz, E. N. Deterministic nonperiodic flow / E. N. Lorenz // J. Atmos. Sci. – 1963. – Vol. 20. – P. 130–141.
2. Rössler, O. E. An equation for continuous chaos / O. E. Rössler // Phys. Lett. A. – 1987. – Vol. 57. – P. 397–398.
3. Sprott, J. C. Some simple chaotic flow / J. C. Sprott // Phys. Rev. E. – 1994. – Vol. 50. – P. R 647–650.
4. Sprott, J. C. Simplest dissipative chaotic flow / J. C. Sprott // Phys. Lett. A. – 1997. – Vol. 228. – P. 271–274.
5. Zhang, Fu. Nonchaotic behaviour in three -dimensional quadratic systems / Fu Zhang, J. Heidel // Nonlinearity. – 1999. – Vol. 10. – P. 1289–1303.
6. Heidel, J. Nonchaotic behaviour in three -dimensional quadratic systems II. The conservative case / J. Heidel, Fu Zhang // Nonlinearity. – 1999. – Vol. 12. – P. 617–633.
7. Zhang, Fu. Chaotic and nonchaotic behavior in three -dimensional quadratic system: 5 -1 dissipative cases / Fu Zhang, J. Heidel // Inter. Journal of Bif. and Chaos. – 2012. – Vol. 22, № 1. – P. 1250010.
8. Грицук, Е. В. К теории нелинейных систем дифференциальных уравнений со свойством Пенлеве / Е. В. Грицук, В. И. Громак // Весці НАН Беларусі. Серыя фіз.-мат. навук. – 2006. – № 3. – С. 25–30.
9. Цегельник, В. В. Аналитические свойства решений семейства нелинейных трехмерных диссипативных систем с хаотическим поведением / В. В. Цегельник // Динамические системы: устойчивость, управление, оптимизация: материалы Междунар. науч. конф. памяти профессора Р. Ф. Габасова, Минск, 5 -10 октября 2021 г. / Белорус. гос. ун-т; редкол.: Ф. М. Кириллова (гл. ред.) [и др.]. – Минск: Изд. центр БГУ, 2021. – С. 194–196.

# АЛГОРИТМ РАБОТЫ УГЛОВОГО ДИСКРИМИНАТОРА РЛС ТОЧНОГО ИЗМЕРЕНИЯ КООРДИНАТ ПРИ ВОЗДЕЙСТВИИ КИМ-ПОМЕХИ

Чигирь И. В.

Учреждение образования «Военная академия Республики Беларусь»

Минск, Республика Беларусь

E-mail: ivan.chigir.a@gmail.com

*Для импульсно-доплеровских радиолокаторов сопровождения разработан алгоритм работы углового дискриминатора при воздействии комбинированной имитирующей и маскирующей помехи.*

## ВВЕДЕНИЕ

Алгоритм защиты от воздействия КИМ-помех предполагает при обнаружении маскирующей составляющей КИМ-помех осуществить ее компенсацию [2] и на фоне остатков компенсации решать задачу обнаружения, измерения параметров ОС. В свою очередь КИМ-помеха является помехой самоприкрытия, т.е. информацию об угловом положении ВЦ несут не только отраженный от цели сигнал, но и все составляющие КИМ-помехи. Энергетическое превосходство КИМ-помехи может обеспечить повышение точности пеленгации сопровождаемой ВЦ. Для повышения точности углового сопровождения ВЦ при постановке с ее борта КИМ-помех необходимо адаптировать угломер ИД РЛС к виду и параметрам такого рода помех.

## 1. ПОСТАНОВКА ЗАДАЧИ

Рассмотрим ситуацию, когда ИД РЛС сопровождает по дальности, радиальной скорости и угловым координатам одиночную ВЦ. В момент времени  $t_0$  с борта ВЦ осуществляется постановка КИМ-помехи. В начальный момент постановки КИМ-помехи ( $t_0$ ) и в течение первой секунды увода, время запаздывания и частота Доплера УП соответствуют времени запаздывания и частоте Доплера ОС  $t_r^{up} = t_r^s, F_d^{up} = F_d^s$  [1]. В ИД РЛС следящее измерение угловых координат осуществляется методом амплитудного мгновенного сравнения (АМС) с суммарно-разностной обработкой [3], при этом формируются одна передающая и четыре приемных диаграммы направленности антенны (ДНА – рисунок 1) с шириной  $\Delta\beta$  в азимутальной и  $\Delta\epsilon$  в угломестной плоскостях.

Сигнал на выходе суммарного канала формируется как сумма сигналов на выходе каналов приема  $f_1(t), f_2(t), f_3(t), f_4(t)$ , формируемых четырьмя парциальными ДНА  $f_{\Sigma}(t) = f_1(t) + f_1(t) + f_1(t) + f_1(t)$ . Сигналы на выходе разностного азимутального  $f_{\Delta\beta}(t)$  и угломестного  $f_{\Delta\epsilon}(t)$  каналов формируются по следующим алгоритмам  $f_{\Delta\beta}(t) = [f_1(t) + f_4(t)] - [f_1(t) + f_1(t)]$  и  $f_{\Delta\epsilon}(t) = [f_1(t) + f_2(t)] - [f_3(t) + f_4(t)]$ . Угловое направление передающей ДНА совпадает с равносигнальным направлением (rsn) и характери-

зуется координатами  $\Delta\epsilon_{rsn}, \Delta\beta_{rsn}$ . Приемные ДНА смещены относительно передающей на заданные углы  $\delta\theta_{\beta}, \delta\theta_{\epsilon}$  и их пространственное положение определяется угловыми координатами  $\beta_{trk}, \epsilon_{trk}$ .

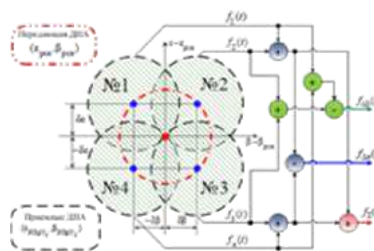


Рис. 1 – Схема расположения передающей, приемных ДНА в картинной плоскости

В ИД РЛС используется когерентное накопление импульсов на всем интервале наблюдения, составляющем  $\approx 5$  мс. В результате время-частотной обработки принятого сигнала в приемнике обзора формируется матрица «дальность-скорость»  $Z$  [3]. Данная матрица представляет собой выходные сигналы  $I$  взаимно расстроенных узкополосных фильтров (УПФ) с учетом квадратичного детектирования, принимаемых в  $J$  каналах однозначного измерения дальности (рисунок 2).

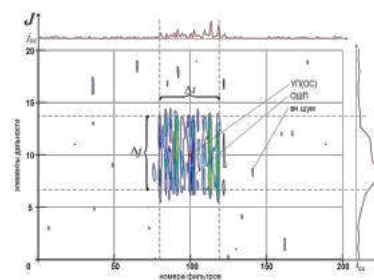


Рис. 2 – Результат обработки принятого сигнала в матрице «дальность-скорость»

Количество элементов разрешения по частоте  $I$  матрицы «дальность-скорость» определяется числом когерентно накапливаемых импульсов  $L_{kn}$ . Когерентное накопление осуществляется на всем интервале наблюдения, поэтому  $I$  соответствует числу импульсов ОС  $L$ . Ширина одного УПФ  $\Delta F_{II}$  определяется временем наблюдения  $T_n$  (числом импульсов ОС  $L$  и периодом



повторения  $T_p$ ) [3].

$$\Delta F_{II} = 1/T_n \quad (1)$$

Количество элементов разрешения по времени запаздывания  $J$  определяется периодом повторения  $T_p$  и интервалом дискретизации  $\Delta t$

$$J = T_p/\Delta t \quad (2)$$

Уводящая по дальности и скорости помеха представляет собой сосредоточенный объект в пространстве времени и частоты соответствующий вектору временных параметров  $\alpha_{up}^T = |t_r^{up}, F_d^{up}|$  (рисунок 2).

В матрице «дальность-скорость»  $Z$  УП занимает элементы  $i_{up}$  и  $j_{up}$ . Протяженная по дальности коррелированная ОШП занимает некоторое количество элементов разрешения по дальности и скорости, т.е. представляет собой пространственно распределенную цель. Значение априорно неизвестного параметра  $\alpha_h^T = |T_0^h, t_r^h, \Delta f_h|$  – длительности одиночного шумового импульса –  $T_0^h$ , времени запаздывания одиночного шумового импульса –  $t_r^h$  и ширины ее спектра частот –  $\Delta f_h$  можно выразить через количество элементов разрешения по частоте и времени запаздывания соответственно

$$T_0^h = \Delta j_h \cdot \Delta t \quad (3)$$

$$t_r^h = \Delta j_h^{min} \cdot \Delta t \quad (4)$$

$$\Delta f_h = \Delta i_h \cdot \Delta F_{II} \quad (5)$$

где  $\Delta i_h = \Delta i_h^{max} - \Delta i_h^{min}$  – количество УПФ занятых ОШП;  $\Delta j_h = \Delta j_h^{max} - \Delta j_h^{min}$  – количество интервалов дискретизации занятых ОШП;  $i_h^{min} (i_h^{max})$  – минимальный (максимальный) номер фильтра занятый ОШП;  $j_h^{min} (j_h^{max})$  – минимальный (максимальный) номер интервала дискретизации занятый ОШП.

Векторному параметру  $\alpha_h$ , с учетом выражений (3) – (5), соответствует

$$\theta_h = |i_h^{min}, i_h^{max}, j_h^{min}, j_h^{max}|^T \quad (6)$$

Элементы вектора  $\theta_h$  подлежат определению в результате решения задачи адаптации [2], реализация оптимального алгоритма обнаружения предполагает использование оценок неизвестных элементов вектора параметров  $\theta_h$ .

## II. АЛГОРИТМ КВАЗИОПТИМАЛЬНОГО УГЛОВОГО ДИСКРИМИНАТОРА ПРИ ВОЗДЕЙСТВИИ КИМ-ПОМЕХ

При постановке с борта ВЦ КИМ-помех алгоритм работы квазиоптимального дискриминатора метода АМС записывается в следующем виде (7), где  $S_\Sigma(t, \Delta\theta_t) = A_\Sigma \cdot S_{\Sigma f}(t) + S_{\Sigma noise}(t)$

– сигнал на выходе устройства когерентной обработки основного (суммарного) канала измерителя;  $S_\Delta(t, \Delta\theta_t) = A_\Delta \cdot S_{\Delta f}(t) + S_{\Delta noise}(t)$  – сигнал на выходе устройства когерентной обработки дополнительного (разностного) канала измерителя;  $Z_{ij}$  – значение квадрата модуля корреляционного интеграла, соответствующие элементам времени и частоты. После суммарно-разностной обработки, для каждого дискрета дальности формируется сигнал ошибки по углам. В пределах КИМ-помехи (маскирующей ее составляющей – ОШП) по дальности берется два отсчета разностных каналов по углам и умножаются на одноименный отсчет суммарного канала и делится на квадрат его модуля. Далее в том же строке дальности проводятся те же самые операции для всех остальных отсчетов дальности (вычисляется сигналы ошибок по углам) и затем усредняется с учетом весов. Тем самым получают уточненные сигналы ошибки по углу места и азимуту в течении одного времени наблюдения. Сигналы ошибки, полученные по результатам нескольких обзоров, поступают на схемы фильтрации и коррекции, где формируются команды для управления силовыми следящими приводами.

## III. ЗАКЛЮЧЕНИЕ

Применительно для ИД РЛС сопровождения разработан алгоритм работы углового дискриминатора при воздействии комбинированных имитирующих и маскирующих помех, отличающийся возможностью оценивать угловые координаты двух видов активных помех с адаптацией к их время-частотной структуре. Его работа позволяет повысить точность пеленгации ВЦ при постановке с ее борта КИМ-помех.

## IV. СПИСОК ЛИТЕРАТУРЫ

1. Van Brunt, Leroy. B. Applied ECM. Vol. 1, 2 / Leroy. B. Van Brunt – EW Engineering Inc, 1972. – 1700 p
2. Чигирь, И. В. Анализ воздействия комбинированной имитирующей и маскирующей помехи на системы сопровождения по дальности и скорости радиолокаторов точного измерения координат / И. В. Чигирь, С. А. Горшков, Н. К. Кузьмичев // Вестн. ВА РБ. – 2018. – № 1 (58). – С. 71–81.
3. Горшков, С. А. Основы радиолокации: конспект лекций. Ч. 3 / С. А. Горшков, С. Ю. Седышев, А. С. Солонар, С. Н. Ярмолик. – Мн: ВА РБ, 2015.

$$D_{AMC}(t, \Delta\theta_t) = \frac{\sum_{i_h^{min}}^{i_h^{max}} \sum_{j_h^{min}}^{j_h^{max}} Z_{ij} \cdot \frac{S_\Sigma(t, \Delta\theta_t) \cdot S_\Delta^*(t, \Delta\theta_t)}{|S_\Sigma(t, \Delta\theta_t)|^2}}{\sum_{i_h^{min}}^{i_h^{max}} \sum_{j_h^{min}}^{j_h^{max}} Z_{ij}} \quad (7)$$

# ПРОТОТИПИРОВАНИЕ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ

Шатилова О. О., Кукин Д. П., Гуревич О. В., Коршикова Д. В.  
Кафедра вычислительных методов и программирования,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: {o.shatilova, kudin, o.gurevich, korshikova}@bsuir.by

*В статье рассматриваются основные этапы проектирования пользовательских интерфейсов игровых приложений, а также психофизические факторы, которые необходимо учитывать при разработке.*

## ВВЕДЕНИЕ

Пользовательский интерфейс (User Interface, UI) – внешний вид продукта, который обеспечивает связь между пользователями и программным продуктом. При проектировании необходимо учитывать, что от степени продуманности UI напрямую зависит степень заинтересованности пользователей продуктом. Чем проще, лаконичней и понятней будет результат, тем выше вероятность, что пользователи останутся довольны продуктом, а, соответственно, продукт будет приносить стабильную прибыль.

### I. ЭТАПЫ ПРОТОТИПИРОВАНИЯ

Основной целью прототипирования является выявление степени актуальности проектируемого приложения. Для этого необходимо выявить целевую аудиторию (ЦА), для чего и как пользователи будут пользоваться им, а также каким образом можно заработать на этом продукте. На этапе определения ЦА нужно подключить эмпатию для четкого определения мыслей, целей пользователей. Для этого можно использовать следующие методы: создание фокус групп, интервьюирование представителей ЦА, создание опросников, нацеленных на определение проблемного поля и оценки потребности в создаваемом продукте.

После определения ЦА следует непосредственно этап прототипирования приложения, который начинается с создания макета. При проектировании мобильного приложения следует учитывать ограниченность экрана, а также потенциально большое количество фичей.

Этап прототипирования состоит из нескольких шагов:

#### 1. Определение структуры и главных функциональных частей интерфейса

Исходя их разбивки игры на логические части, описанные в концепт-документе, в первую очередь, разрабатывается техническое задание на мокапы, которые представляют собой визуализацию отдельных частей приложения. На этом этапе определяется количество и местоположение кнопок навигации, их размеры, наличие или

отсутствие надписей к ним, стили переходов между экранами.

#### 2. Сборка прототипа, определение стилистики

На этом шаге создаются прототипы игровых экранов, то есть визуализируется техническое задание мокапов. Одновременно прорабатывается стилистика, потому что при совмещении отдельных частей приложения может возникнуть несогласованность по графической и концептуальной составляющим.

#### 3. Опрация прототипа и отрисовка превью интерфейса

На этом этапе более тщательно прорабатывается графика с учетом референсов, для того чтобы привести все части графического интерфейса к цельному восприятию.

#### 4. Доработка интерфейса и добавление анимаций

#### 5. Анализ полученного результата, устранение недостатков

На выходе получается интерфейс игрового приложения, однако для максимизации эффекта в процессе разработки и реализации прототипа кроме самого процесса проектирования необходимо подключать знания психофизиологии - науки, изучающей механизмы психических процессов, состояния и поведения.

### II. ПСИХОФИЗИОЛОГИЧЕСКИЕ АСПЕКТЫ РАЗРАБОТКИ ПРОТОТИПА

С точки зрения теории систем, человек является информационной системой, подобной компьютеру, и получает информацию через различные каналы: визуальный, звуковой, тактильный. Основным, как правило, является визуальный канал, поэтому в процессе разработки прототипа основное внимание стоит уделять аспектом именно графического восприятия.

Для повышения читабельности и облегчения выбора нельзя перегружать экран информацией. Существует правило «7+-2», которое сформулировал американский ученый психолог Джорж Миллер, согласно которому, если человеку предоставляется выбор из более большого числа вариантов, то он избегает решать такую задачу. Таким образом, количество актив-

ных элементов, расположенных на экране должно быть в пределах до 7, с учетом размера дисплея выходного устройства. Рекомендуется группировать информацию в группы и предоставлять пользователю выбирать информацию по категориям.

С точки зрения запоминаемости, а значит и узнаваемости, на первом месте стоят цифры, затем цвета и только потом буквы. Это обусловлено эволюционными процессами, которые предполагают, что мозг пытается экономить ресурсы, и для решения поставленных задач пытается найти оптимальные пути решения.

В 2014 году Нобелевскую премию в области медицины получила группа ученых (Джон О'Киф, Университет Лондона, Мэй-Бритт Мозер и Эдварду Мозеру, Норвежский университет естественных наук и технологии), которые провели исследование мозга и открыли клетки, которые отвечают за позиционирование, позволяющее организму ориентироваться в пространстве, точно запоминать местоположение и прокладывать маршруты. Таким образом, у людей в мозге есть «сетка», позволяющая навигировать внутри пространства, и она масштабируема – это наша геометрическая память. Если человек запоминает, где располагается объект, то у него не будет необходимости задействовать другие участки мозга, для того, чтобы запомнить его цвет и форму.

При прототипировании важно помнить о корректной группировке объектов, для того, чтобы облегчить поиск нужной информации (см. рис.1).



Рис. 1 – Примеры группировки объектов (корректная и некорректная)

Для привлечения внимания очень хорошо работают контрасты форм, цветов, размеров, а также движения. При разработке мобильных приложений отлично может сыграть прямой контраст цвета и размера для акцентирования внимания пользователя на важных для продавца приложения элементах UI. Также неплохо работает контраст движения, если, например, иконку с монетизацией заставить вращаться, она не может быть проигнорирована мозгом пользователя.

Также дизайнеры UI/UX рекомендуют тщательно подбирать шрифты для надписей, а для мобильных приложений желательно надписи заменять пиктограммами.

### III. Выводы

Проектирование UI – достаточно большая и трудоемкая часть разработки приложения. От качества реализации этого этапа во многом зависит успешность вопроса в целом, потому что даже самая отличная идея может быть уничтожена нечитаемым и непривлекательным экраном приветствия или перегруженным экраном настроек приложения, которые в совокупности могут отвернуть потенциального клиента от использования продукта. В связи с этим, разработчикам приложений стоит уделять должное внимание проектированию UI, чтобы успешно продвигать свой проект на рынке приложений.

### IV. СПИСОК ЛИТЕРАТУРЫ

1. Раскин, Д. Интерфейс. Новые направления в проектировании компьютерных систем / Д. Раскин, – 2005. – 273 с.
2. Психофизиология графических интерфейсов [Электронный ресурс] / Психофизиология графических интерфейсов. – Режим доступа: <http://https://tilda.education/courses/web-design/psychophysiology/>. – Дата доступа: 30.10.2021.
3. Особенности создания интерфейса для мобильной игры [Электронный ресурс] / Особенности создания интерфейса для мобильной игры. – Режим доступа: <https://dtf.ru/gamedev/44813-osobennosti-sozdaniya-interfeysa-dlya-mobilnoy-igry>. – Дата доступа: 30.10.2021.

# РАЗРАБОТКА КОМПЛЕКСА ВИРТУАЛЬНЫХ ЛАБОРАТОРНЫХ РАБОТ (КВЛР) «ТЕХНОЛОГИИ СООРУЖЕНИЯ ПЕРЕХОДОВ И СЛОЖНЫХ УЧАСТКОВ МАГИСТРАЛЬНЫХ ГАЗОНЕФТЕПРОВОДОВ»

Ширшова М.А., Дружинская Е.В., Фатхиев Т.И., Хасанов Р.Р.

Институт цифровых систем, автоматизации и энергетике, кафедра вычислительной техники и инженерной кибернетики, факультет трубопроводного транспорта, кафедра промышленных трубопроводных систем, Уфимский государственный нефтяной технический университет  
Уфа, Российская Федерация  
E-mail: shirshovamasha@gmail.com

*Сооружение сложных участков и переходов через естественные и искусственные препятствия магистральных газонефтепроводов является достаточно опасным, ответственным и трудно моделируемым процессом.*

Тенденции развития современного мира требуют разрабатывать виртуальные и цифровые комплексы, позволяющие проектировать и рассчитывать инженерные сооружения и моделировать различные условия их эксплуатации. В связи со сложной санитарно-эпидемиологической обстановкой в мире учебные заведения во многих странах были вынуждены перейти к дистанционным форматам обучения.

Разрабатываемое программное обеспечение (ПО) «Комплекс виртуальных лабораторных работ «Технология сооружения переходов и сложных участков магистральных газонефтепроводов»» позволяет обучающимся самостоятельно обучаться в рамках расширения направления подготовки. Продукт предоставляет возможность не только моделирования различных технологических процессов, аварийных ситуаций и параметров прокладки трубопровода, но и визуализации этапов сооружения и ввода в эксплуатацию трубопроводных систем транспорта нефти, нефтепродуктов и газа.

ПО строится на основе педагогических принципов электронного обучения, реализуемых с помощью технических средств современных информационных технологий.

Для проектирования программной реализации комплекса проведено теоретическое исследование технологий сооружения сложных участков и переходов магистральных газонефтепроводов:

- выделены основные этапы;
- выполнен подбор необходимой техники;
- произведены необходимые технические расчеты задействованных параметров;
- составлены тестовые задания для контроля правильности выполнения технологии.

На основании результатов исследования составлено техническое задание (ТЗ) на разработку программного средства.

При составлении ТЗ использовались учебные материалы и учебно-методическая литература по направлению подготовки специалистов

в области проектирования и сооружения газонефтепроводов (21.03.01 Нефтегазовое дело). В основу теоретического материала легли

- отраслевые стандарты;
- научно-техническая документация;
- нормативные документы такие, как своды правил (СП) и государственные стандарты (ГОСТы).

В качестве среды разработки выбрана Unity от Unity Technologies с использованием технологии Unity3d, благодаря чему появилась возможность использования трехмерного пространства. Весь функциональный расчет реализован на стандартном языке программирования C#. Однако из-за необходимости наличия детальных 3D-моделей было решено обратиться к Blender от Blender Foundation. С его помощью создано несколько видов техники (см. рис. 1, рис. 2), труб и их бетонирования.



Рис. 1 – Модель обычного экскаватора с обратной лопатой

В программном комплексе реализуется технология укладки трубопроводов на болотистых участках, в многолетнемерзлых грунтах и на сильно пересеченной (гористой) местности. Каждый этап считается завершенным только при правильном подборе или расчете параметров, то есть переход на следующий шаг регулируется программой.

После введения исходных данных предоставляется выбор техники для выполнения земляных работ. При правильном выборе техника визуализируется и запускается 3D-демонстрация

процесса работы. В это время обучающийся производит расчет параметров трубопровода.



Рис. 2 – Модель трубоукладчика

Программа производит расчёт толщины стенки по формуле (1) и сверяет его со значением, вычисленным и введённым обучающимся в соответствующее поле. Реализовано распознавание вводимых вещественных значений вне зависимости от используемого разделителя.

$$\delta = \frac{n * p * D_n}{2 * (R_1 + n * p)}, \quad (1)$$

где  $\delta$  – расчетная толщина стенки, мм;  $D_n$  – наружный диаметр трубопровода, мм;  $p$  – рабочее давление трубопроводе, МПа;  $R_1$  – расчетное сопротивление растяжению, МПа;  $n$  – коэффициент надежности по нагрузке.

Однако, чтобы перейти на следующий шаг, полученное значение должно удовлетворять трем условиям (2), (3) и (4). При вводе верных значений появляется возможность прокладки рассчитанного трубопровода с последующим его бетонированием и очисткой. Деятельность комплекса представлена диаграммой на рис. 3.

$$0 < \frac{p * (D_n - 2 * \delta)}{2 * \delta} < \frac{m * R_2^H}{0.9 * k_n}, \quad (2)$$

где  $R_2^H$  – нормативное сопротивление сжатию, МПа;  $k_n$  – коэффициент надежности по трубопроводу;  $m$  – коэффициент условий работы.

$$| -\alpha_t * E * \Delta_{t^x} + \mu + \sigma_{kc}^H - \frac{E * D_n}{1000} | > \frac{\psi_1 * m * R_2^H}{0.99 * k_n} \quad (3)$$

$$0 < -\alpha_t * E * \Delta_{t^t} + \mu + \sigma_{kc}^H + \frac{E * D_n}{1000} < \frac{m * R_2^H}{0.9 * k_n}, \quad (4)$$

где  $\mu$  – коэффициент Пуассона (0,3);  $\alpha_t$  – коэффициент линейного расширения металла трубы (0,000012);  $-$  модуль Юнга (206000 МПа);  $\Delta_{t^x}$

– положительный температурный перепад;  $\Delta_{t^t}$  – отрицательный температурный перепад;  $\psi_1$  – коэффициент двухосного напряженного состояния металла труб, для  $\Delta_{t^t}$  равен 1 (5);  $\sigma_{kc}^H$  – нормативные кольцевые напряжения, МПа (6).

$$\psi_1 = \sqrt{1 - 0,75 * \left( \frac{\sigma_{kc}^H * 0,9 * k_n}{m * R_2^H} \right)^2 - 0,5 * \frac{\sigma_{kc}^H * 0,9 * k_n}{m * R_2^H}} \quad (5)$$

$$\sigma_{kc}^H = \frac{p * (D_n - 2 * \delta)}{2 * \delta} \quad (6)$$

Разработанное приложение может быть использовано в профориентационной деятельности, в организации образовательного процесса в технических вузах по направлениям подготовки, связанных с проектированием, сооружением, эксплуатацией и ремонтом магистральных трубопроводов, а также на курсах переподготовки работников нефтегазовой промышленности.

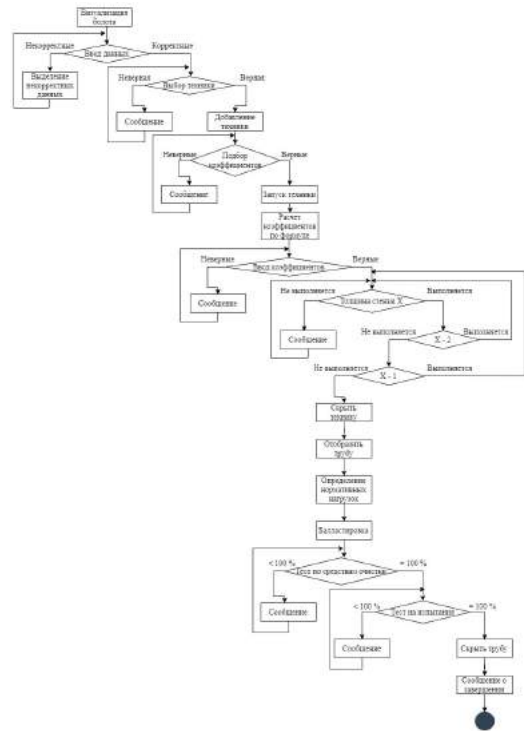


Рис. 3 – Диаграмма деятельности комплекса

#### Литература

1. Крец В.Г. Сооружение и эксплуатация газонефтепроводов и газонефтехранилищ [Текст]: учебное пособие / В.Г. Крец, А.В. Шадрин, Н.А. Антропова. – Томск: Изд-во ТПУ, 2012. – 386 с.
2. Челнокова Е.А. Проектирование и разработка дистанционного курса [Текст] / Е.А. Челнокова, А.С. Челноков, К.А. Максимова, И.А. Лапшина, И.Е. Барбина // Инновационная экономика: перспективы развития и совершенствования. – 2020. – № 2. – С. 65-69.
3. Официальный сайт Blender [Электронный ресурс]. – Режим доступа: <https://www.blender.org/>. – Дата доступа: 19.10.2021.
4. Официальный сайт Unity [Электронный ресурс]. – Режим доступа: <https://unity.com/ru>. – Дата доступа: 19.10.2021.

# ПОСЛЕДОВАТЕЛЬНЫЙ АЛГОРИТМ СИГНАЛЬНО-ТРАЕКТОРНОГО РАДИОЛОКАЦИОННОГО РАСПОЗНАВАНИЯ С АДАПТАЦИЕЙ К УГЛАМ ПРОСТРАНСТВЕННОЙ ОРИЕНТАЦИИ ОБЪЕКТА

Ярмолик С. Н., Зайко Е. В., Свинарский М. В., Леонович А. С.  
Кафедра автоматика, радиолокации и приемо-передающих устройств,  
Военная академия РБ  
Минск, Республика Беларусь  
E-mail: zaiko.eugene@mail.ru, mechislav1993@gmail.com

*В рамках доклада представлен вариант устройства последовательного распознавания при совместном использовании сигнальных и траекторных признаков с адаптацией к УПО объекта. Для формирования апостериорных плотностей используется численный метод Монте-Карло.*

## ВВЕДЕНИЕ

Факт принадлежности наблюдаемого объекта к определенному классу (типу) устанавливается в процессе решения задачи радиолокационного распознавания (РЛР). Информация о классе (типе) наблюдаемого объекта используется при решении широкого спектра задач на различных уровнях управления. В современных условиях вооруженного противоборства подразделений войск противовоздушной обороны и средств воздушного нападения противника особую актуальность приобретает вопрос обеспечения высокой эффективности решения задачи РЛР в реальном масштабе времени.

## I. ОСНОВНАЯ ЧАСТЬ

Совершенствование алгоритмов РЛР достигается путем совместного использования: комплексирования и комбинирования классификационных признаков [1], адаптации классификационных признаков к условиям наблюдения объектов [4], последовательных процедур принятия решения [2 С. 35-44]. Для решения задачи РЛР в качестве классификационных признаков широко используют радиолокационные портреты (РЛП) объектов, а также траекторные признаки, характеризующие закономерности движения объектов [1]. При этом стоит отметить, что существенное влияние на вид и параметры РЛП оказывают углы пространственной ориентации (УПО) наблюдаемого объекта [3]. В рамках доклада представлен вариант устройства последовательного распознавания при совместном использовании сигнальных и траекторных признаков с адаптацией к УПО объекта [2, 3, 4]. Решение о классе объекта или о продолжении наблюдения принимается согласно решающему правилу: для  $(n + 1) > K$  :  
если  $\begin{cases} Z(\mathbf{f}_{n+1}|A_k) > Z(\mathbf{f}_{n+1}|A_l), l = \overline{1, M}, l \neq k \\ Z(\mathbf{f}_{n+1}|A_k) > \ln \left( \sum_{g=1, g \neq k}^M e^{Z(\mathbf{f}_{n+1}|A_g)} \right) + \\ + \ln \left( \frac{1 - \Pi_{M+1}}{\Pi_{M+1}} \right) \end{cases}$ , то  $A_k^*$ , иначе  $A_{M+1}^*$   
для  $(n + 1) = K$  : если  $Z(\mathbf{f}_{n+1}|A_k) > Z(\mathbf{f}_{n+1}|A_l)$ ,  
 $l = \overline{1, M}, l \neq k$ , то  $A_k^*$ , где  $Z(\mathbf{f}_{n+1}|A_k) =$

$z_\eta(\mathbf{f}_{n+1}|A_k) + z_\xi(\mathbf{f}_{n+1}|A_k)$  - обобщенное значение логарифма правдоподобия при совместном использовании сигнальных и траекторных признаков в  $k$ -м канале обработки с первого по текущий  $(n+1)$ -й контакт;  $\Pi_{M+1}$  - стоимость принятия решения о продолжении наблюдения;  $z_\eta(\mathbf{f}_{n+1}|A_k) = \ln \left( L_\eta(\mathbf{f}_{n+1}|A_k) \right)$  - логарифм коэффициента правдоподобия по траекторным признакам в  $k$ -м канале обработки с первого по текущий  $(n+1)$ -й контакт;  $z_\xi(\mathbf{f}_{n+1}|A_k) = \left( \sum_{r=1}^{n+1} z_\xi(\mathbf{f}_r|A_k) \right)$  - логарифм отношения правдоподобия по сигнальным признакам в  $k$ -м канале с первого по текущий  $(n+1)$ -й контакт;  $K$  - значение номера контакта, при котором принимается решение о классе объекта без возможности продолжения наблюдения;  $\mathbf{f}_{n+1} = \{\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_{n+1}\}$  - совокупность принятых реализаций отраженного сигнала с первого по текущий  $(n+1)$ -й контакт. Структурная схема устройства РЛР представлена на рисунке 1. Стоит отметить, что особенность представленного алгоритма состоит в использовании численного метода Монте-Карло для аппроксимации апостериорной плотности вероятности УПО  $p(\hat{\psi})$  и плотности вероятности оценки траекторных признаков  $p(\hat{V}_r)$  наблюдаемого объекта (рисунок 2) [5]. Эффективность функционирования предложенного алгоритма оценивалось методом математического моделирования. В качестве показателей качества использовались вероятности правильного и ложного распознавания класса объекта, а также длительность процедуры принятия решения. Анализировались объекты трех классов: «Крылатая ракета», «Истребитель», «Бомбардировщик». Для анализируемых объектов моделировались прямолинейные траектории полета, отличающиеся параметрами их пространственного перемещения. В качестве сигнальных классификационных признаков использовались дальномерные радиолокационные портреты, а в качестве траекторных - высотно-скоростные характеристики наблюдаемых объектов. Результаты математического моделирования в виде зависи-

мостей вероятностей правильного ( $D$ ) и ложного ( $F$ ) распознавания, а также средней длительности процедуры принятия решения ( $\bar{N}$ ) от отношения сигнал/шум на выходе фильтра сжатия для трех распознаваемых классов представлены на рисунке 3.

## II. ЗАКЛЮЧЕНИЕ

Полученные результаты иллюстрируют эффективность представленного алгоритма последовательного РЛР при совместном использовании сигнальных и траекторных признаков с адаптацией к УПО наблюдаемого объекта. Использование при аппроксимации апостериорных плотностей вероятности УПО наблюдаемого объекта и оценки траекторных признаков численного метода Монте-Карло позволяет практически реализовать последовательный алгоритм классификации объектов в современных и (или) перспективных РЛС.

## III. СПИСОК ЛИТЕРАТУРЫ

1. Горшков, С. А. Сочетание сигнальных и траекторных признаков в задачах измерения-распознавания / С. А. Горшков, А. С. Солонар // Прикладная радиоэлектроника. — 2009. — Т. 8, №4. — С. 497–500.
2. Храменков, А. С. Последовательное радиолокационное распознавание с минимизацией риска принятия решения о продолжении наблюдения и о классе объекта : дис. ... канд. техн. наук : 05.12.14 / А. С. Храменков. — Минск, 2017. — 194 с.
3. Ярмолик, С. Н. Методика синтеза байесовского алгоритма распознавания радиолокационных объектов с адаптацией к их углам пространственной ориентации / С. Н. Ярмолик [и др.]. // Вест. Воен. акад. Респ. Беларусь. -- 2018. — №4. — С. 53–66.
4. Ярмолик, С. Н. Синтез оптимального устройства последовательного сигнально-траекторного распознавания с адаптацией к пространственной ориентации наблюдаемого объекта / С. Н. Ярмолик [и др.]. // Вест. Воен. акад. Респ. Беларусь. -- 2020. — №3. — С. 80–91.
5. Ярмолик, С. Н. Адаптация к пространственной ориентации объектов в задаче сигнально-траекторного распознавания классов / С. Н. Ярмолик [и др.]. // Информационные технологии и системы 2020 : материалы междунар. науч. конф., Минск, 18 нояб. 2020 г. / редкол.: Л. Ю. Шилин [и др.]. — Минск: БГУИР, 2020. — С. 213–214.

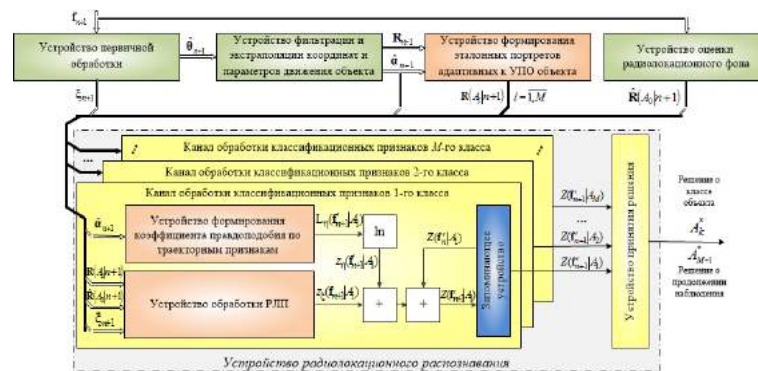
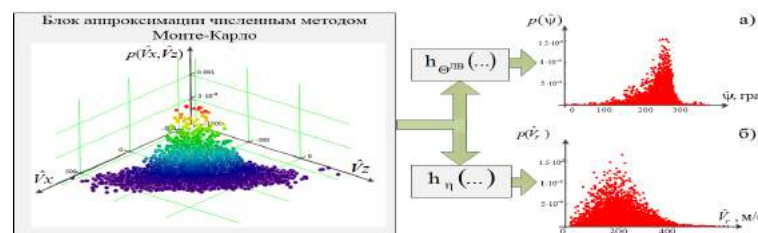


Рис. 1 – Структурная схема последовательного устройства радиолокационного распознавания по сигнальным и траекторным признакам с адаптацией к УПО объекта



а) плотность вероятности угла курса; б) плотность вероятности радиальной скорости

Рис. 2 – Пример аппроксимации плотностей вероятности численным методом Монте-Карло

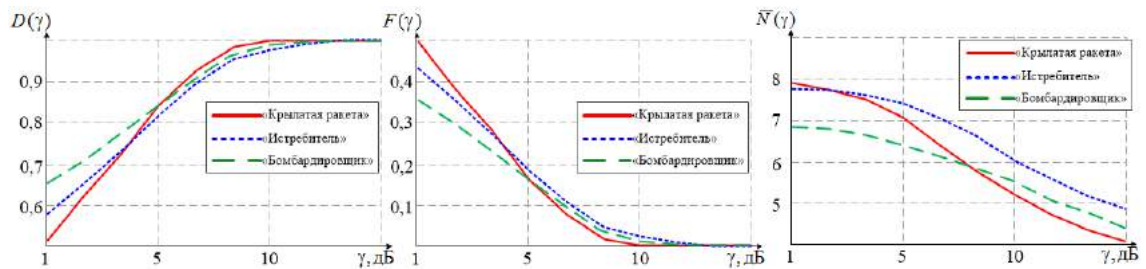


Рис. 3 – Зависимости показателей качества последовательного радиолокационного распознавания от отношения сигнал/шум

# ENTERPRISE RESOURCE PLANNING FOR STARTUPS IN IRAQ

Hussein A. T., Gurinovich A. B.

Department of Informatics, Belarusian State University of Informatics and Radioelectronics

Minsk, Republic of Belarus

E-mail: Rm77hussin@gmail.com, gurinovich@bsuir.by

*refers to a type of software that organizations use to manage day-to-day business activities such as accounting, procurement, project management, risk management and compliance, and supply chain operations. A complete ERP suite also includes enterprise performance management, software that helps plan, budget, predict, and report on an organization's financial results. In Iraq, enterprise resource planning is used in most governmental and private companies and institutions. this Book For ERP systems (<https://www.pdfdrive.com/enterprise-resource-planning-erp-d42992753.html>).*

## INTRODUCTION

ERP which stands for Enterprise Resource Planning is a modular software system designed to integrate the main functional areas of a company's business processes into one integrated system. ERP software standardizes, simplifies, and integrates business processes including finance, human resources, procurement, distribution, and other departments. In 1990, Gartner coined the term ERP to describe the evolution of material requirements planning (MRP) and manufacturing resource planning (MRP II) as both expanded beyond manufacturing into other parts of the company, usually finance and human resources.

### I. DEFINITION OF ERP SYSTEMS

ERP systems typically include the following characteristics:

- An integrated system
- Operates in (or near) real time
- A common database that supports all the applications
- A consistent look and feel across modules
- Installation of the system with elaborate application/data – integration by the Information Technology (IT) department, provided the implementation is not done in small steps [21]
- Deployment options include: on-premises, cloud hosted, or SaaS.

### II. IMPLEMENTATION

ERP's scope usually implies significant changes to staff work processes and practices. Generally, three types of services are available to help implement such changes: consulting, customization, and support. Implementation time depends on business size, number of modules, customization, the scope of process changes, and the readiness of the customer to take ownership for the project. Modular ERP systems can be implemented in stages. The typical project for a large enterprise takes about 14 months and requires around 150 consultants. Small projects can require months;

multinational and other large implementations can take years. Customization can substantially increase implementation times.

### III. CHARACTERISTICS

ERP systems typically include the following characteristics:

- An integrated system
- Operates in (or near) real time
- A common database that supports all the applications
- A consistent look and feel across modules
- Installation of the system with elaborate application/data integration by the Information Technology (IT) department, provided the implementation is not done in small steps
- Deployment options include: on-premises, cloud hosted, or SaaS.

### IV. CONNECTIVITY TO PLANT FLOOR INFORMATION

ERP systems connect to real-time data and transaction data in a variety of ways. These systems are typically configured by systems integrators, who bring unique knowledge on process, equipment, and vendor solutions. Direct integration—ERP systems have connectivity (communications to plant floor equipment) as part of their product offering. This requires that the vendors offer specific support for the plant floor equipment their customers operate. Database integration—ERP systems connect to plant floor data sources through staging tables in a database. Plant floor systems deposit the necessary information into the database. The ERP system reads the information in the table. The benefit of staging is that ERP vendors do not need to master the complexities of equipment integration. Connectivity becomes the responsibility of the systems integrator Enterprise appliance transaction modules (EATM)— These devices communicate directly with plant floor equipment and with the ERP system via methods supported by the ERP system. EATM can employ a staging table, web services, or system-specific program



interfaces (APIs). An EATM offers the benefit of being an off-the-shelf solution. Custom-integration solutions—Many system integrators offer custom solutions. These systems tend to have the highest level of initial integration cost, and can have a higher long term maintenance and reliability costs. Long term costs can be minimized through careful system testing and thorough documentation. Custom-integrated solutions typically run on workstation or server-class computers.

#### V. HOW DOES AN ERP SYSTEM WORK?

The main purpose of an ERP system is to increase organizational efficiency of an organization by managing and improving how company resources are utilized. Improving and/or reducing the number of resources necessary without sacrificing quality and performance are keys to effectively improving business growth and profitability. ERP software has the ability to collect and compare metrics across departments and provide a number of different reports based on roles or specific user preferences. The data collected makes finding and reporting on data faster and gives a complete view of business performance with complete insights on how resources are being spent. ERP synchronizes reporting and automation by reducing the need to maintain separate databases and spreadsheets that would have to be manually merged to generate reports. This combined data collection and reporting offers valuable insight, such as where to cut costs and streamline processes, providing the information to make real-time business decisions.

#### VI. WHAT INDUSTRIES CAN BENEFIT FROM ERP?

ERP software can be used in any industry to help a business become more efficient. It provides an effective communication tool that can manage information between internal and external departments, assist with daily activities to manage projects, track adherence to guidelines, and handle day-to-day intricacies that come with running a business. Because the roots of enterprise planning software are closely associated with manufacturing, there are robust manufacturing ERP solutions that cater to a variety of specific industries. ERP software systems are very diverse and are key parts of many industries, including but not limited to:

- Industrial Machinery and Components
- Construction and Home Improvement
- Electronics and Technology
- Aerospace and Defense
- Healthcare, Pharmaceutical and Life Sciences
- Agribusiness, Farming and Agriculture

#### VII. WHAT IS THE BUSINESS VALUE OF ERP?

ERP systems are used to help businesses of all sizes overcome challenges—from small businesses to massive enterprises. Early business practices may no longer keep up with growing demand and require more efficient business tools, like ERP, to effectively manage a business' systems and resources. ERP software systems provide many benefits to the health and growth of a business.

#### VIII. CONCLUSION

There are as many reasons for successful ERP implementations as there are for failed projects. However, success seems to often be measured by whether or not the project came in on time and under budget. Whereas, fully utilizing the system to achieve improved business practices appears to be ignored. Performance measures must be developed and standardized to give organizations a clearer picture of the benefits derived from Enterprise Resource Planning implementation. Much has been written about and learned from some well-publicized successes and failures in ERP implementations. Some of it has even been directly contradictory. However, most agree on some basic rules:

- Establish the business processes prior to selecting the software.
- Staff the project team with members of the user community in addition to IT staff.
- Develop an implementation plan and stick to it.
- Train the users thoroughly on the process changes and flow of information in addition to the actual software.
- The project doesn't end with "go-live", but must be continually monitored.

#### IX. REFERENCES

1. Eric L. Keller, "Lessons Learned", *Manufacturing Systems*, v17, iss.11, pp. 44-50.
2. R. Michael Donovan, "No magic cure will fix all ERP ills."

# NIGERIAN BANKS DATA ANALYTICS

Williams E., Gurinovich A. B.

Department of Informatics , Belarusian State University of Informatics and Radioelectronics

Minsk, Republic of Belarus

E-mail: williamselvis0007@gmail.com , gurinovich@bsuir.by

*This research aims at identifying the factors that play a significant role in the forming of customer behavior toward the brand or the organization. After previewing the literature of marketing, the researcher found that the human resources management activities are the main contributors in retaining the brand's name and the organization's name stuck in the customer's mind. The study confirms that CRM is a complex concept requiring appropriate business process and integrated systems. In addition, the study demonstrates the relevance of the need for effective leadership, sourcing, communication and evaluation within CRM strategies.*

## INTRODUCTION

Customer relationship management (CRM) is a management strategy that unites information technology with marketing. It originated in the United States in the late 1990's, and, to date, has been accepted in a significant number of companies worldwide. On the other hand, some people have negative opinions of CRM; such views hold that it is difficult to implement successfully and that its cost/benefit performance is low, among others. Customer relationship management (CRM) is a broadly recognized, widely-implemented strategy for managing and nurturing a Company's interactions with customers, clients and sales prospects. It involves using technology to organize, automate, and synchronize business processes, principally sales activities, but also those for marketing, customer service, and technical support. The overall goals are to find, attract, and win new clients, nurture and retain those the company already has, entice former clients back into the fold, and reduce the costs of marketing and client service.

### I. MARKET ENVIRONMENT

In the early 20th century in the United States, demand outpaced supply to the extent that companies concentrated on selling as many products as possible. Suppliers focused on product development, manufacturing capacity, and securing distribution outlets, without regard to their consumers. They did not pay much attention to who bought their products or what their customers needed. They used classic marketing tactics, i.e., mass marketing – primarily print and broadcast advertising, mass mailings, and billboards. By the middle of the 20th century, however, the economy had matured to a point where consumers had the power of choice because supply had outstripped demand. The era of the passive consumer was coming to an end. Companies began to find out who their customers were, what they wanted, and how they could be satisfied.

## II. CUSTOMER MANAGEMENT

There are many definitions of customer relationship management presented by, either scholars or practitioners or researchers, and from various perspectives. (Parvatiyar and Sheth, 2004) have included the four dimensions of customer relationship management in their definition, where they defined it as "a comprehensive strategy and process of acquiring, retaining, and partnering with selective customers to create superior value for the company and the customers". While marketers have defined CRM "CRM is a comprehensive and marketing strategy; that integrates technology, process and all business activities around the customer (Anton and Hoek, 2002).

## III. RESEARCH PROBLEM

Formulated the problem of this research by the following statement: "CRM can not be implemented, through single organizational activity". This problem will be discussed through the following questions: 1. What are the activities of an indirect effect on the implementation of CRM strategy? 2. Which organizational activity has the biggest role in the implementation of CRM strategy? 3. What is the organizational activity which has the biggest role in the successful implementation of CRM Strategy?

## IV. RESEARCH IMPORTANCE

Customer relationship management is very important for any business enterprise, because it is considered the main generator of any business success (Pharma CRM report, 2008). And the importance of this research is due to what was pointed out by (Roh, 2005), that many organizations have employed CRM system to improve decision making process, to provide goods and services in highest quality and to build long-term relationship with customer which benefits both the customer and the organization (Ng Uyen, et al, 2007).

## V. RESEARCH OBJECTIVES

This research aims to achieve the following:  
1. Determine the activity of a larger role in the implementation of CRM strategy, with a focus on the role of human resource management and the role of information technology. 2. Explain how to use CRM by organizations in achieving customer satisfaction and loyalty, which lead to retaining the name of the organization and its products, either goods or services, in the customer's mind.

## VI. RESEARCH HYPOTHESES

Prior to the hypotheses of this research, we must recall that the implementation of the CRM strategy serves as a change done by organizations seeking the best, the most important requirement of the success of the change is to create and catalyst climate for change. Therefore, the hypotheses of this research will be as follows: a. Ha1: Human resource practices will have a significant impact on organizational climate. b. Ha2: The organizational climate will be significantly related to customer service indices. c. Ha3: The overall satisfaction plays a mediating role between transaction-specific satisfaction and attitudinal loyalty. d. Ha4: Customer loyalty significantly linked to his insisting on dealing with the organization, but not others, and his preference for its products and services.

## VII. BENEFIT IMPLEMENTATION

The most important benefits are achieving overall customer satisfaction that leads to gain customer loyalty (Keh and Lee, 2006). Some researchers have argued that the two benefits are the driver for many of the related benefits such as; increased sales volume, improve service quality, increase profits (Zineldin, 2005). CRM benefits can be divided, to tangible benefits and intangible benefits. According to (Chen and Chen, 2004) the tangible benefits are: 1. The increase in profitability. 2. Speed in the investment of time. 3. The growth of labor productivity. 4. Reduce marketing costs. 5. Increase customer relation rates. 6. Achieving higher return on marketing's investments.

## VIII. IDENTIFY CATEGORIZE LEADS

A CRM system can help you identify and add new leads easily and quickly, and categorize them accurately. By focusing on the right leads, sales can prioritize the opportunities that will close deals, and marketing can identify leads that need more nurturing and prime them to become quality leads.

With complete, accurate, centrally held information about clients and prospects, sales and marketing can focus their attention and energy on the right clients.

## IX. REDUCE COSTS

CRM can be quick and easy to implement. A cloud-based system doesn't need special

installation, and there's no hardware to set up, keeping IT costs low and removing the headache of version control and update schedules.

Generally, cloud-based CRM systems are priced on the number of users who access the system and the kinds of features needed. This can be very cost-effective in terms of capital outlay, and is also extremely flexible — enabling you to scale up and add more people as your business grows. Salesforce is flexible in terms of functionality, too — you're not paying for any features that are not useful to you.

## X. OFFER BETTE SUPPORT

Today's customers expect fast, personalized support, at any time of day or night. A CRM system can help you provide the high-quality service that customers are looking for. Your agents can quickly see what products customers have ordered, and they can get a record of every interaction so they can give customers the answers they need, fast.

## XI. CONCLUSION

As mentioned earlier this paper has two objectives. These objectives are to examine the activity of a larger role in the implementation of CRM strategy, with a focus on the role of human resource management and the role of information technology. It shows a strong relationship it today's business environment between HR and use CRM strategy to measure performance on customer in order to enhance customer satisfaction. Human Resources practices were found to significantly and positively affect performance. In addition the use of customer performance measures partially mediates the relationship between HR and performance. The study confirms that CRM is a complex concept requiring appropriate business process and integrated systems. in addition, the study demonstrates the relevance of the need for effective leadership, sourcing, communication and evaluation within CRM strategies. According to the second objective of the study there are basic conditions one showed consider when starting a CRM implementation system one of the basic conditions is that organizational climate as defined customer satisfaction and loyalty has to understand the business process inadequate information in the system regarding self-renewing, innovative, agile and aligned with customer needs and expectations.

## XII. REFERENCES

1. - AL-Damen, Rula (2006) Developing an Excellence Assessment Model for Jordanian Businesses, Jordan: Amman Arab University for Graduate Studies.
2. Anton, J. and Hoeck, M. (2002). E-Business Customer Service. Santa Monica, CA: The Anton Press.
3. this Book For CRM systems <https://www.amazon.com/Handbook-Relationship-Marketing-Atul-Parvatiyar/dp/0761918108>



Секция «Автоматизированные  
системы обработки информации»

Section «Automated information  
processing systems»

# СИСТЕМА МАРКИРОВКИ И ПРОСЛЕЖИВАЕМОСТИ ТОВАРОВ, КАК МЕТОД БОРЬБЫ С КОНТРАФАКТНОЙ ПРОДУКЦИЕЙ

Архипенко Я. С., Пилипенко В.В.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь

E-mail: arkhipenko.yana.serg@gmail.com, vitaliy.law@gmail.com

*Обеспокоенность безопасностью потребительских свойств продукции в последнее время стала высокой. Значительные кризисы в сфере производства продуктов питания и непродовольственных товаров в Европе за последние пять лет дал повод средствам массовой информации уделить много внимания таким проблемам, как диоксиды, пестициды, содержание вредных веществ в составе товаров, а также большое количество контрафактов.*

## ВВЕДЕНИЕ. ОСНОВНЫЕ ПОНЯТИЯ

Прослеживаемость – новое направление в программе безопасности продуктов питания, производства одежды, обуви и других товаров. Идентификация и управление информацией выступает в качестве средства быстрого и точного обнаружения или отзыва продуктов.

Система прослеживаемости товаров – механизм контроля и наблюдения за состоянием продукции на всех стадиях движения по цепочке «производство – поставка – реализация». Система прослеживаемости позволяет своевременно обнаружить и устранить возможные риски, влияющие на безопасность конечной продукции для потребителя.

### Принципы маркировки товаров:

- нанесение индивидуального средства идентификации на упаковку продукции;
- контроль движения маркированных упаковок по цепочке поставок, осуществляемый операторами системы прослеживаемости;
- сбор данных о сбыте маркированной продукции и их передача регулирующим и правоохранительным органам.

### СРЕДСТВА ВНЕДРЕНИЯ СИСТЕМЫ МАРКИРОВКИ И ПРОСЛЕЖИВАЕМОСТИ

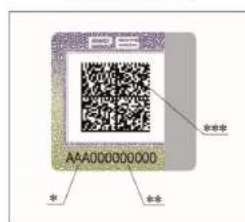
Для внедрения прослеживаемости необходимо на всех этапах цепи поставки внедрять решения, дающие возможность определить происхождение, местоположение, маршрут движения продукта или партии продуктов. Эффективная система прослеживания должна позволять отследить продукты вниз или вверх по цепи поставки, т.е. ответить на вопросы «Где?» находится интересующий Вас объект и «Откуда?» он пришел, т.е. определить происхождение объекта.

С точки зрения управления информационными процессами внедрение систем прослеживаемости в цепи поставок требует от всех вовлеченных торговых партнеров систематического объединения физического потока материалов, полу-

фабрикатов и готовых продуктов с информационным потоком, описывающим их.

Качественный товар, выпущенный добросовестным производителем, должен быть промаркирован в соответствии с требованиями страны-производителя. На территории Республики Беларусь, Российской Федерации и стран Европы существуют определенные национальные операторы системы маркировки товаров. Интеграторы системы маркировки и прослеживаемости товаров предлагают различное программное обеспечение для взаимодействия с национальными операторами. Основной функционал ПО: возможность получить особые Унифицированные контрольные знаки (документ с определенной степенью защиты, предназначенный для маркировки товаров – рис. 1) для нанесения на продукцию, присвоение групповой упаковке продукции кода прослеживаемости, формирование отчета о нанесении унифицированных контрольных знаков и агрегации продукции, отправка отчетов о нанесении и агрегации оператору маркировки.

Образец унифицированного контрольного знака  
(формат 18 x 18 мм)



\* Серия (переменная информация).  
\*\* Номер (переменная информация).  
\*\*\* Уникальная последовательность символов в машинчитаемой форме, представленная в виде двумерного штрихового кода DataMatrix, включающая в себя переменную информацию, в том числе серию и номер унифицированного контрольного знака.

Рис. 1 – Унифицированный контрольный знак - DataMatrix код

Потребитель может проверить качество товара путем сканирования DataMatrix кода. В Республике Беларусь используется система "Электронный знак на территории Российской Федерации – "Честный знак". Если производитель добросовестный, он обязан отправить отчет

о нанесении национальному оператору системы маркировки, и вся продукция производителя будет иметь статус «доступен для продажи». Если при сканировании унифицированного контрольного знака появляется информация «товар не может находиться в продаже», значит отчет о нанесении не был отправлен, возможно товар является контрафактом, но с большей долей вероятности по каким-то причинам произошла ошибка во время отправки отчета.



Рис. 2 – Результат сканирования DataMatrix кода приложением "Честный знак продукция" промаркирована верно



Рис. 3 – Результат сканирования DataMatrix кода приложением "Честный знак продукция" промаркирована неверно, товар не может находиться в продаже



Рис. 4 – Результат сканирования DataMatrix кода приложением "Электронный знак продукция" промаркирована верно, товар может находиться в продаже



Рис. 5 – Результат сканирования DataMatrix кода приложением "Электронный знак продукция" промаркирована не до конца, унифицированный контрольный знак лишь выдан системой, отчет о нанесении еще не поступил национальному оператору маркировки

### ПРИМЕНЕНИЕ В ИНТЕРНЕТ-МАГАЗИНАХ

Унифицированные контрольные знаки больше пользы могут принести при введении их обязательной фиксации в интернет-магазинах. Наряду с информацией о стоимости, внешнем виде, особых параметрах продукта (цвет, габариты, вес и т.д.) в описании продукта можно добавить отдельное поле с четкой фотографией DataMatrix кода (или поле с текстовым отображением кода). При оформлении заказа покупатель сможет просканировать данный код, в общей базе продукции увидеть, что данный код единственный в своем роде, принадлежит именно производителю и продукт не является подделкой. После доставки товара к месту назначения, покупатель сможет сверить код на полученном товаре с кодом, который был просканирован ранее при заказе и сохранен в системе. в каждом унифицированном контрольном знаке обязательно должен быть зашифрован уникальный идентификатор производителя (код, набор символов, текст, ключ и др.)

Таким образом можно попробовать снизить количество контрафактных товаров.

### I. СПИСОК ЛИТЕРАТУРЫ

1. Государственная информационная система маркировки товаров "Электронный знак"[Электронный ресурс]: РУП "Издательство "Белбланкавыд"/ Минск, 2021. – режим доступа: <https://datamark.by/>
2. GS1 Russia. The Global Language of Business [Электронный ресурс]: GS1 AISBL 2020. / Москва. – режим доступа: <http://www.gs1ru.org/traceability/>

# ПРИМЕНЕНИЕ МАШИННОГО ОБУЧЕНИЯ ДЛЯ УВЕЛИЧЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ СУБД PostgreSQL

Белошедов Е. С., Гуринович А. Б.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: 2595118@mail.ru, gurinovich@bsuir.by

*Машинное обучение занимается поиском скрытых закономерностей в данных, что позволяет получить исключительные результаты. С этим связан растущий рост интереса к этой теме в ИТ-сообществе. Распознавание речи и отсканированных документов, почтовые машины всё это функционирует на базе машинного обучения. Исследуются способы применения методов машинного обучения для увеличения производительности СУБД.*

## ВВЕДЕНИЕ

SQL представляет собой декларативный язык. То есть пользователь указывает только на те операции, которые должны быть проделаны с данными. За выбор способа выполнения этих операций отвечает СУБД. Можно разделить процесс поиска оптимального плана на две части. Во-первых, нужно уметь оценивать стоимость любого плана — количество ресурсов, необходимых для его выполнения. В случае, когда на сервере не выполняются другие задачи и запросы, оцениваемое время выполнения запроса прямо пропорционально количеству потраченных на него ресурсов. Поэтому можно считать, что стоимость плана — это его время выполнения в некоторых условных единицах. Во-вторых, требуется выбрать план с минимальной оценкой стоимости. Легко показать, что число планов растёт экспоненциально с увеличением сложности запроса, поэтому нельзя просто перебрать все планы, оценить стоимость каждого и выбрать самый дешёвый. Для поиска оптимального плана используются более сложные алгоритмы дискретной оптимизации: динамическое программирование по подмножествам для простых запросов и генетический алгоритм для сложных [1].

### I. ПОИСК СУЩЕСТВУЮЩЕГО ОПТИМАЛЬНОГО ПЛАНА ВЫПОЛНЕНИЯ ЗАПРОСА

В PostgreSQL для плана предсказываются две стоимости: стоимость запуска (start-up cost) и общая стоимость (total cost). Стоимость запуска показывает, сколько ресурсов план потратит до того, как выдаст первую запись, а общая стоимость — сколько всего ресурсов потребуется плану для выполнения. Эта задача разделяется на две подзадачи. Сначала для каждой вершины плана предсказывается, сколько кортежей будет отобрано в ней. Затем на основе этой информации оценивается стоимость выполнения каждой вершины, и, соответственно, всего плана. В случаях, когда предположение о независимости условий не выполняется, модель

PostgreSQL работает не совсем корректно. Следующие способы позволяют преодолеть эту проблему. Первый способ заключается в построении многомерных гистограмм. Проблема этого способа заключается в том, что с увеличением размерности, многомерная гистограмма требует экспоненциально растущее количество ресурсов для сохранения той же точности. Поэтому приходится ограничиваться гистограммами небольшой размерности (2-8 измерений). Отсюда следует вторая проблема этого метода: нужно каким-то образом понять, для каких пар столбцов имеет смысл строить многомерные гистограммы, а для каких необязательно. Чтобы решить эту проблему, требуется либо хороший администратор, который будет изучать планы ресурсоемких запросов, определять корреляции между столбцами и вручную указывать, какие гистограммы нужно достроить, либо программное средство, которое с помощью статистических тестов попытается найти зависимые друг от друга столбцы. В настоящий момент существуют патчи, позволяющие использовать в PostgreSQL многомерные гистограммы, но в них администратору требуется вручную задавать, для каких столбцов эти многомерные гистограммы должны быть построены [2].

### II. ИСПОЛЬЗОВАНИЕ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ОЦЕНКИ ВЫБОРОЧНОСТИ

Альтернативный подход — это применение машинного обучения для нахождения совместной выборочности нескольких условий. Как уже говорилось выше, машинное обучение занимается поиском закономерностей в данных. Данные — это набор объектов. В данном случае объектом является совокупность условий в одной вершине плана. По этим условиям и их маргинальным выборочностям требуется предсказать совместную выборочность. Наблюдаемыми признаками вершины плана будут являться маргинальные выборочности всех её условий. Необходимо считать эквивалентными между собой все условия, отличающиеся только в константах. Мож-



но рассматривать данное допущение как типичный прием машинного обучения — hashing trick — примененный для уменьшения размерности пространства. Предполагается, что вся необходимая для предсказания информация о константах условия содержится в его маргинальной выборочности. Можно показать это строго для простых условий вида  $a < \text{const}$ : здесь по выборочности условия возможно восстановить значение константы, то есть потери информации не происходит (см. рис. 1).

| Selectivity | users.age > const | users.city = const | messages.sender_id = users.id |
|-------------|-------------------|--------------------|-------------------------------|
| 0.25        | 0.25              | -                  | -                             |
| 0.23        | 0.25              | 0.6                | -                             |
| 0.3         | 0.5               | 0.6                | -                             |
| 0.0005      | -                 | 0.5                | 0.001                         |
| ...         | ...               | ...                | ...                           |

Рис. 1 – Задача машинного обучения

Можно заметить, что, если теперь использовать линейную регрессию, то в качестве частного случая получается текущую модель PostgreSQL.

### III. МЕТОДЫ ТЕСТИРОВАНИЯ

Для тестирования различных подходов был использован бенчмарк TPC-H. В качестве простых регрессоров были использованы следующие методы:

- Гребневая линейная регрессия и стохастический градиентный спуск. Этот метод хорош тем, что позволяет использовать динамическое обучение (online learning), поэтому не требует хранить никаких наблюдаемых объектов.
- Множество гребневых линейных регрессий и стохастический градиентный спуск.
- Множество гребневых линейных регрессий и аналитическое решение методом Гаусса.

Ответы регрессора являются входными значениями для оптимизатора, который ищет оптимальный план. Наблюдаемые объекты (исполняемые планы), являются выходными значениями оптимизатора. Поэтому наблюдаемые объекты зависят от ответов регрессора. Такие системы с обратной связью намного сложнее для изучения, чем системы, в которых регрессор не влияет на окружающую среду. Именно в этих терминах аналитическое решение методом Гаусса является нестабильным — оно быстро обучается, но предлагает более рискованные решения, поэтому в целом система работает хуже [3].

### IV. ИСПОЛЬЗОВАНИЕ МАШИННОГО ОБУЧЕНИЯ

После детального изучения линейной модели было обнаружено, что она недостаточно полно описывает данные. Поэтому наилучшие результаты из опробованных нами методов показал kNN. Существенный минус этого метода заключается в необходимости сохранения в памяти всех объектов с последующей организацией быстрого поиска по ним. Существенно улучшить эту ситуацию можно, используя алгоритм отбора объектов. Идея наивного алгоритма отбора объектов: если предсказание на объекте достаточно хорошее, то запоминать этот объект не нужно. Следующим этапом является тестирование системы для алгоритма kNN (см. рис. 2).

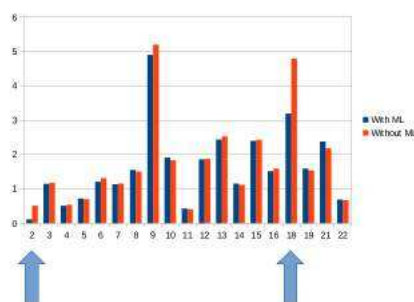


Рис. 2 – Результаты машинного обучения

Можно заметить, что предложенный подход в самом деле ускоряет время работы СУБД.

### V. ЗАКЛЮЧЕНИЕ

Текущий алгоритм гарантирует, что в тех планах, к которым сходится алгоритм, предсказания выборочности будут правильными. Однако это не гарантирует глобальной оптимальности выбранных планов. Поиск глобально оптимальных планов или хотя бы лучшего локального оптимума — это отдельная задача. С внедрением машинного обучения возможно прервать выполнение плана, в котором были допущены серьезные ошибки в предсказании выборочности, учесть полученную информацию и выбрать новый лучший план для выполнения. В большинстве случаев новый план будет существенно отличаться от предыдущего. В процессе работы СУБД меняются данные и типичные запросы. Следовательно, данные, полученные в прошлом, могут быть уже неактуальными.

### VI. СПИСОК ЛИТЕРАТУРЫ

1. Mangrum, J. M. The evaluation and management of bradycardia / J. M. Mangrum, J. P. DiMarco // N. Engl. J. Med. – 2000. – Vol. 342, № 10. – P. 703–709.
2. Neil Conway. Доклады про внутреннее устройство PostgreSQL.
3. Bruce Momjian. Explaining the Postgres Query Optimizer.

# МОДЕЛИРОВАНИЕ ВИБРАЦИОННЫХ СИГНАЛОВ С ЗАДАНЫМИ ХАРАКТЕРИСТИКАМИ

Бранцевич П. Ю., Лапицкая Н. В., Леванцевич В. А.  
Кафедра программного обеспечения информационных технологий,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: branc@bsuir.edu.by, lapan@bsuir.by, lvn@bsuir.by

*Анализируются проблемы проведения метрологической аттестации и поверки виброизмерительного оборудования. Рассмотрена структурная схема системы поверки. Предложен подход моделирования вибрационных сигналов, основанный на фильтре Калмана и программное средство для его реализации. Описана методика проведения поверки и результаты работы программного средства.*

## ВВЕДЕНИЕ

Состояние производственного объекта характеризуется многими параметрами, имеющими различную физическую природу. При эксплуатации механизмов и агрегатов одними из важнейших параметров, отражающим их техническое состояние, являются интенсивность и спектральный состав вибрации.

Для контроля вибрационных сигналов используют различные стационарные и мобильные измерительно-вычислительные комплексы и приборы [1]. Для обеспечения достоверности работы контрольно-измерительного оборудования, его периодически подвергают метрологической аттестации или поверке. Поверка является сложной и трудоемкой процедурой, поэтому весьма актуальной является задача создания программно-аппаратных средств для автоматизации процесса поверки виброизмерительных приборов и вибропреобразователей.

### I. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

На рис. 1 приведен пример структурной схемы системы поверки виброизмерительного оборудования [2].

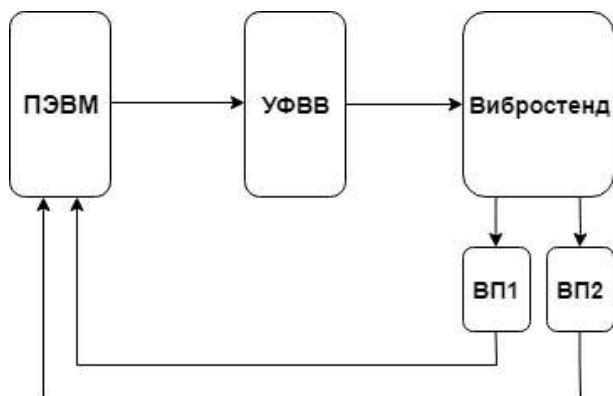


Рис. 1 – Структурная схема системы поверки виброизмерительного оборудования

Поверяемый вибропреобразователь ВП2 закрепляется на рабочем столе вибростенда. Вычислительная машина по специальным алгоритмам формирует управляющие команды, которые

поступают на устройство формирования вибрационного воздействия (УФВВ). УФВВ формирует вибрационный сигнал с заданной амплитудой и частотным спектром, который усиливается и поступает на вибростенд, что приводит к возвратно-поступательному движению стола вибростенда.

Эталонный вибропреобразователь ВП1 формирует сигнал, отражающий реальную частоту и амплитуду перемещения стола вибростенда. Сигналы с датчиков ВП1 и ВП2 поступают в ЭВМ, где оцифровываются и обрабатываются с помощью специализированного программного обеспечения, после этого можно сделать вывод о характеристиках поверяемого датчика. Так как основным элементом вибростенда является катушка возбуждения, обладающая собственной индуктивностью и емкостью, то его амплитудно-частотная характеристика неравномерна и, как следствие, движение стола вибростенда не будет соответствовать параметрам заданного сигнала вибрационного воздействия. Поэтому стоит задача формирования такого сигнала, который позволит компенсировать неравномерность АЧХ вибростенда.

### II. РЕАЛИЗАЦИЯ

Для компенсации неравномерности АЧХ вибростенда предлагается подход, основанный на рекурсивном фильтре Калмана. Математически алгоритм реализации фильтра Калмана в общем случае выражается формулой [3]

$$X_k = X_{k-1} + K_k (Z_k - X_{k-1}), (1)$$

где  $X_k$  - текущее значение формируемой величины на выходе фильтра Калмана;

$X_{k-1}$  - значение формируемой величины, полученное на предыдущем этапе с учетом предсказания;

$Z_k$  - измеренное в текущий момент значение измеряемой величины с датчика;

$K_k$  - коэффициент Калмана.

Коэффициент Калмана зависит от соотношения среднеквадратичной ошибки предсказания и ошибки измерения и с каждым шагом вы-

числения стабилизируется к определенному значению. С целью адаптации фильтра Калмана для коррекции амплитуд спектральных составляющих (1) было преобразовано в следующее выражение:

$$U_{j,i} = U_{j,i-1} \left( 1 + \alpha \left( \frac{A_{z,i}}{A_{izm,i-1}} - 1 \right) \beta \right).$$

где  $U_{j,i}$  - амплитуда сигнала на выходе устройства формирования вибрационного воздействия

$A_{z,j}$  - заданная амплитуда  $j$ -ой составляющей спектра вибрационного воздействия;

$A_{izm,i-1}$  - амплитуда измеренной  $j$ -ой составляющей вибрационного воздействия на  $(i-1)$ -ой итерации;

$\alpha_i = \alpha_{i-1} K_{osl}$  - коэффициент Калман,  $\alpha \leq 1$ ;

$K_{osl}$  - коэффициент ослабления, задается пользователем;

$\beta$ -коэффициент, учитывающий нелинейность выходной характеристики вибростенда,  $\beta=1$ ;

$j, i$  - номер гармоники и номер итерации соответственно.

Условием завершения алгоритма коррекции спектра является выполнение либо условия  $|A_{z,i} - A_{izm,j,i}| \leq \delta A_{z,j}$  для всех спектральных составляющих, где  $\delta$  - погрешность установки спектральной составляющей, либо  $i > i_{gr}$ , где  $i_{gr}$  - граничное значение числа итераций коррекции. После выхода на заданный общий уровень вибрационного воздействия, осуществляются действия по определению характеристик поверяемого ВП.

При таком подходе методика поверки ВП выглядит следующим образом:

1. Определяются значения параметров для поверки: диапазон частот; общий уровень мощности воспроизводимого вибросигнала в точке установки эталонного вибропреобразователя; равномерный спектр воспроизводимого вибросигнала; значения контрольных частот.

2. С помощью системы поверки виброизмерительного оборудования и программного обеспечения поверки[2] в точке крепления образцового ВП воспроизводится вибросигнал с заданными характеристиками.

3. Определяются характеристики принимаемого сигнала от поверяемого ВП и рассчитываются значения основных параметров: коэффициент преобразования в диапазоне частот, коэффициенты преобразования на контрольных частотах, АЧХ для коэффициента преобразования в заданном диапазоне частот.

4. Пункты 2-3 повторяются для нескольких значений мощности.

Разработаны алгоритмы и программное средство, поверки вибропреобразователей. Скриншоты работы программы приведены на рис. 2-4.

На рис. 2 показана форма полигармонического сигнала вибрационного воздействия, формирующего равномерный спектр. На рис. 3. приведен реальный, неравномерный спектр колебаний в точке крепления вибропреобразователя до коррекции, а на рис. 4 равномерный спектр, полученный в той же точке, после коррекции программой поверки.

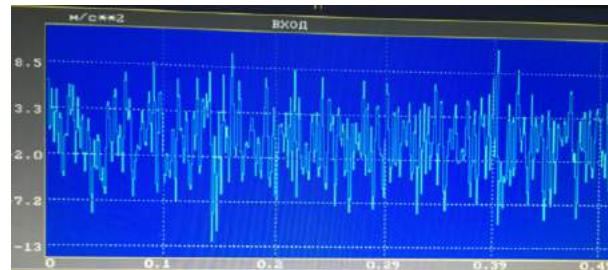


Рис. 2 – Исходный сигнал вибрационного воздействия

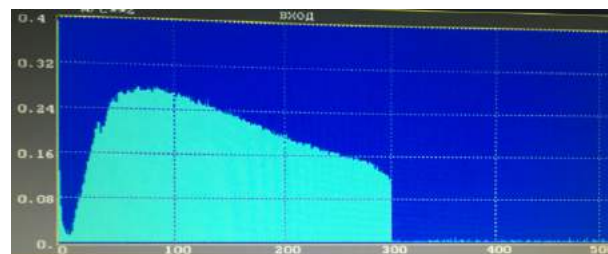


Рис. 3 – Неравномерный спектр сигнала в точке крепления вибропреобразователя до коррекции

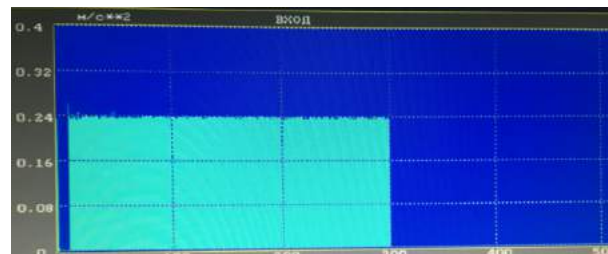


Рис. 4 – Равномерный спектр сигнала в точке крепления вибропреобразователя после коррекции

### III. ВЫВОДЫ

Применение разработанной методики и программного средства позволяет значительно сократить время поверки и повысить точность определения значений поверяемых параметров.

### IV. СПИСОК ЛИТЕРАТУРЫ

1. Неразрушающий контроль. Справочник. Том 7. Книга Вибродиагностика /Ф.Я. Балицкий и др. М.: Машиностроение, 2005. – 485 с.
2. Бранцевич, П.Ю. Компьютерные вибродиагностические системы / П.Ю. Бранцевич, С.Ф. Костюк, Е.Н. Базылев, В.Э. Базаревский // Междунар. науч.-техн. конф., приуроченная к 50-летию МРТИ-БГУИР: материалы конф. – Минск : БГУИР, 2014. – Ч. 1, – с. 430–431
3. Васильев, В.Н., Компьютерная обработка сигналов в приложении к интерферометрическим системам./В.Н. Васильев, И.П. Гуров.– СПб: БХВ, 1998.–22 – 250

# ОБ ОДНОЙ МОДЕЛИ ИНТЕРНЕТ-БИЗНЕСА

Герман Ю. О.

Факультет компьютерных сетей и систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: juliagerman@tut.by

*Представлены задачи бизнес-аналитики в web. Одна из списка задач – определение цены на подержанные товары (рассматриваются автомобили). Клиент web-магазина подержанных автомобилей располагает определенной суммой денег и списком критериев для выбора подходящего автомобиля. Исходя из этой информации, система подбирает автомобиль с указанием его цены в рамках указанной суммы, максимально соответствующий системе критериев пользователя. Описанный подход является достаточно общим и может быть использован при решении соответствующего класса бизнес-задач.*

## ВВЕДЕНИЕ

Web является богатым источником прикладных задач, объединяемых в рамках фреймворка, известного как бизнес-аналитика. На нижнем уровне имеют место следующие задачи:

- выбор бизнеса (сути деятельности);
- оценка риска инноваций (открытия (профитности) бизнеса);
- управление ценами;
- прогнозирование объема продаж;
- планирование закупок
- и др.

Более сложные проблемы объединяются в рамках Data Mining и включают

- принятие решений в условиях неопределенности;
- оценка риска банкротства;
- построение модели бизнес-активности;
- интеллектуализация функционала;
- человеко-машинное решение задач
- и т.д.

Целью данного сообщения является описание математической модели для определения цены подержанных товаров (например, автомобилей, стиральных машин, ноутбуков и т.п.). Обычно пользователь располагает определенной суммой денег и не очень отчетливо понимает, на какую цену ему следует ориентироваться. У пользователя есть набор собственных критериев, значениям которых, по его мнению, должен удовлетворять приобретаемый товар. Предлагаемая аналитика позволит пользователю выяснить, какую цену следует заплатить за покупку. Описываемый подход предполагает наличие web-консалтинговой службы, обеспечивающей необходимый математический сервис. Мы будем рассматривать задачу покупки автомобиля. В качестве литературных источников с теоретическим материалом по проблеме отметим [1-3].

## I. ОПИСАНИЕ ЗАДАЧИ

В качестве критериев при определении цены на подержанный автомобиль будем рассматривать: **(К1)** срок эксплуатации; **(К2)** про-

бег; **(К3)** мощность двигателя. Этим критериям должны быть присвоены приоритеты  $w_i$ . Матрица приоритетов является одним из ключевых объектов метода иерархий Т.Саати. В нашем случае она может иметь, например, следующий вид.

|           | <b>К1</b> | <b>К2</b> | <b>К3</b> |
|-----------|-----------|-----------|-----------|
| <b>К1</b> | 1         | 3         | 2         |
| <b>К2</b> | 1/3       | 1         | 1/2       |
| <b>К3</b> | 1/2       | 2         | 1         |

Интерпретация числовых значений в ячейках матрицы такая: 1 - критерии равноценны; 2 или 3 - незначительное превосходство одного критерия над другим; 3-5 превосходство одного критерия над другим; 6-8 существенное превосходство и 9-10 - подавляющее превосходство одного критерия над другим. Далее отыскиваем приведения чисел в каждой строке и извлекаем корень, степень которого равна числу критериев. Обозначим полученные величины  $prod_i$ . Приоритеты  $w_i$  критериев определяем как

$$w_i = \frac{prod_i}{\sum_{i=1}^3 prod_i}$$

Далее должны быть построены функции  $F_i$  полезности по каждому критерию. Опуская подробности, пусть эти функции будут известны. Таким образом для каждого автомобиля  $x$  в базе данных можно определить интегральную оценку  $I(x)$  функции выбора. Мы считаем далее, что база данных по автомобилям содержит достаточно большое число наименований автомобилей и их характеристик. Полагаем, что цены на проданные автомобили известны. В противном случае цена устанавливается продавцом. Продавец также может использовать механизм Саати для формирования продажной цены. Продажная цена может определяться по формуле

$$P_k = P \cdot I_k,$$

где  $P$  представляет цену нового автомобиля данной марки (класса) с подобными техническими параметрами. При указанных исходных данных требуется найти в базе данных автомобиль

(-или), в наибольшей степени соответствующий критериям покупателя и располагаемой им суммой денег.

## II. ОПИСАНИЕ МЕТОДА

Поставленная задача может быть сведена к задаче интерполяции функции с одной переменной. В качестве переменной выступает значение интегрального критерия  $I$  выбора автомобиля (у каждого автомобиля в базе данных свое значение этого критерия). В силу проведенного рассмотрения задачи для каждого проданного автомобиля  $k$  в базе имеется значение интегрального критерия  $I_k$  и цена продажи  $P_k$ . Поскольку автомобилей может быть достаточно много (сотни/тысячи), то система реализует следующий алгоритм.

1. Пересчитывает интегральные функции выбора для автомобилей в соответствии со значениями критериев, введенных пользователем.
2. Отбирает множество (несколько десятков) автомобилей, цены на которые удовлетворяют располагаемой клиентом сумме. При этом желательно, чтобы значения интегральных критериев выбранных автомобилей варьировали в достаточно широких пределах.
3. Строит интерполяционный многочлен  $Poly$ , описывающий значение цены в зависимости от значения интегрального критерия выбора.
4. По цене, выставленной пользователем, находит значение интегрального критерия автомобиля по графику  $Poly$ . Для этого находит точку на оси ординат ( $Y$ ) и откладывает от нее перпендикуляр до пересечения с линией графика (точек пересечения может быть несколько). От точки пересечения с графиком опускает перпендикуляр на ось абсцисс ( $X$ ) для получения значения интегрального критерия  $I$  автомобиля-кандидата на покупку.
5. По найденному значению  $I$  отыскивает в базе данных автомобиль с характеристиками, в наибольшей степени соответствующими  $I$ .

Для построения интерполяционного многочлена можно использовать аналитический язык Python. В языке Python можно выполнять интерполяцию с помощью полиномов: линейных, квадратичных, кубических. Получим линейную и кубическую интерполяцию с помощью следующих методов

```
f1 = interp1d(x, y, kind = 'linear')
f2 = interp1d(x, y, kind = 'cubic')
```

Переменные  $f1$  и  $f2$  представляют собой массивы значений функции интерполяции. Фраг-

ментарно скрипт для интерполяции представлен так, как показано ниже

```
import numpy as np from scipy import interpolate
from scipy.interpolate import interp1d
f1 = interp1d(x, y, kind = 'linear')
f2 = interp1d(x, y, kind = 'cubic')
xnew = np.linspace(0, 4, 30)
plt.plot(x, y, 'o', xnew, f(xnew), '-', xnew, f2(xnew),
'_')
plt.legend(['data', 'linear', 'cubic', 'nearest'], loc =
'best')
plt.show()
```

Данный скрипт не только определяет значения функции интерполяции, но и рисует ее график. Отметим, что необходимо выполнять проверку точности интерполяционной функции (этот вопрос мы здесь не рассматриваем). Имеется еще одна техника интерполирования на основе сплайнов (сплайны – те же полиномы, например полиномы второй или третьей степени).

## III. ЗАКЛЮЧЕНИЕ

Рассмотренная выше задача является одной из типичных задач web-бизнеса. Как видим, ее решение сочетает методы многокритериальной оценки и выбора альтернатив и технику интерполяции. В случае больших массивов данных (Big Data) возникает еще одна проблема - разбиения этих массивов на небольшое число кластеров, причем в каждом кластере содержится некоторое подмножество автомобилей, близких по характеристикам. Поэтому в кластерах следует выбирать центроиды (типичных представителей) и строить интерполяцию на основе этих типичных представителей. Другой возможный подход состоит в вычислениях на выборке (скажем, в пределах 15% от общего числа автомобилей). Эта техника основана на выборочном контроле и также представляет интерес. Имеет место проблема независимости критериев в методе Саати. Есть формальный критерий независимости, но он является достаточно жестким. Более простой вариант проверки независимости состоит в том, чтобы фиксировать значение одного из критериев и проверять, является ли выбор значения второго критерия обусловлен выбранным значением первого. Наличие такой обусловленности свидетельствует о зависимости критериев.

## IV. СПИСОК ЛИТЕРАТУРЫ

1. Francis, A. Business Mathematics and Statistics / A. Francis, // South-Western Cengage Learning, England.–660p.
2. Saati, T.L. Decision making with the analytic hierarchy process / T. L. Saati // Int. J. Services Sciences, Vol. 1, No. 1, 2008 .
3. German, O. Expert systems / O. V. German // Minsk, BSUIR, 2008. – p.90 (in Russian).

# ПРИМЕНЕНИЕ МЕТОДА ВЕКТОРИЗАЦИИ ДЛЯ АНАЛИЗА РУССКОЯЗЫЧНОЙ ТЕКСТОВОЙ ИНФОРМАЦИИ

Гоглев И. В.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: ivangoglev1998@gmail.com

*Рассматривается реализация вопросно-ответной системы на основе векторизации*

## ВВЕДЕНИЕ

На сегодняшний день существует большое множество вопросно-ответных систем (QA-систем). Вопросно-ответная система – это информационная система, являющаяся гибридом поисковых, справочных и интеллектуальных систем, которая использует естественно-языковой интерфейс.[1] Различают общие QA-системы и узкоспециализированные QA-системы. В то же время развились системы и методы обработки информации на естественном языке. Для создания простой QA-системы предлагается использовать метод векторизации предложений.

## I. АНАЛИЗ РУССКОЯЗЫЧНОЙ ТЕКСТОВОЙ ИНФОРМАЦИИ НА PYTHON

Для анализа текстов на русском языке используется библиотека `rumorphy2`. `Rumorphy2` – морфологический анализатор, разработанный на языке программирования Python. Выполняет лемматизацию и анализ слов, способен осуществлять склонение по заданным грамматическим характеристикам слов. Работает со словарём `OpenCorpora`, а для незнакомых слов строит гипотезы[2]. Ниже представлен пример работы данного анализатора.

```
In[7]: import rumorphy2
In[8]: rumorphy2.MorphAnalyzer().parse('книга')
```

Рис. 1 – Пример использования библиотеки `rumorphy2` для морфологического анализа

Для каждого разбора анализатор вычисляет скоринг[2]. Скоринг разбора представляет собой условную вероятность тега  $P(tag|word)$ . Она вычисляется на основе частот словаря `OpenCorpora` для всех неоднозначных слов со снятой неоднозначностью.[2] Для каждого подобного слова оценивается его частота в корпусе и количество сопоставлений его тега. В результате получаем отсортированный по убыванию скоринга массив объектов класса `Parse`. Для слова «книга» анализатор выдаёт ответ следующего вида (рис.2).

```
[Parse(word='книга',
tag=OpenCorporaTag('NOUN,inan,femn sing,nomn'),
normal_form='книга',
score=1.0,
methods_stack=((DictionaryAnalyzer(),
'книга', 44, 0),))]
1
```

Рис. 2 – Результат морфологического анализа

Поле `tag` содержит морфологическую информацию о введённом слове (таблица 1)

Таблица 1 – Теги морфологического анализа

| Тег  | Смысл тега                     |
|------|--------------------------------|
| NOUN | имя существительное            |
| inan | неодушевлённое(inanimate)      |
| femn | женский род(feminine)          |
| sing | единственное число(singular)   |
| nomn | именительный падеж(nominative) |

Также предусмотрен вывод морфологической информации на русском языке (рис.3).

```
In[12]: rumorphy2.MorphAnalyzer().parse('книга')[0].tag_cyr_rep
Out[12]: 'СУШ,неод,жр,ед,им'
```

Рис. 3 – Морфологический анализ на русском языке

`MorphAnalyzer` предоставляет возможность получить нормальную форму слова (именительный падеж и единственное число для существительных) (рис. 4).

```
In[14]: rumorphy2.MorphAnalyzer().parse('книга')[0].normal_form
Out[14]: 'книга'
```

Рис. 4 – Нормальная форма слова

## II. ПРИМЕНЕНИЕ МЕТОДА ВЕКТОРИЗАЦИИ ДЛЯ ОТВЕТА НА ВОПРОС ПО ТЕКСТУ

Перед векторизацией необходимо провести предварительную обработку текстовых данных. Текст преобразуется в массив строк (предложений) и каждое слово в строке приведено к нормальной форме. Стоп-слова (слова, которые не придают особого значения предложению) удаляются из каждого предложения(рис. 5).

```
In[15]: sentences = [
...: 'Он играть поле',
...: 'Он играть парни футбол',
...: 'Игра футбол закончить',
...: 'Дима Вася собирать цветы'
...:]
In[16]: questions = ['Кто играть футбол']
```

Рис. 5 – Предварительная обработка текстовых данных

После предварительной обработки формируем датасет (массив предложений текста и вопроса) (рис. 6)

```
In[17]: sentences+questions
Out[17]:
['Он играть поле',
 'Он играть парни футбол',
 'Игра футбол закончить',
 'Дима Вася собирать цветы',
 'Кто играть футбол']
```

Рис. 6 – Сформированный датасет

Далее производим векторизацию с помощью функции `CountVectorizer` из библиотеки `sklearn` (рис. 7).

```
In[21]: from sklearn.feature_extraction.text import CountVectorizer
In[22]: dataset=sentences+questions
In[23]: vectorizer = CountVectorizer(input=dataset)
In[24]: snts_vectors = vectorizer.fit_transform(dataset)
In[25]: vectors = snts_vectors.toarray()
In[26]: vectors
Out[26]:
array([[0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0],
 [0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0],
 [0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0],
 [1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1],
 [0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0]])
```

Рис. 7 – Векторизация

Полученный результат содержит информацию о взаимосвязях между предложениями (в том числе и их подобии)[3]

Далее разбиваем результат на массив векторов предложений текста и вектор вопроса. После сравниваем векторы предложений с вектором вопроса. Предложение с наиболее близким вектором является ответом на вопрос. (рис. 8)

```
In[34]: print('q: ', questions[0])
...: if max_zero == 0:
...: print('a: ', 'Этого нет в тексте')
...: else:
...: print('a: ', sentences[resp])
...:
q: Кто играть футбол
a: Он играть парни футбол
```

Рис. 8 – Результат работы

### III. ВЫВОДЫ

Библиотека `rumorhу2` и метод векторизации позволяет создать простую вопросно-ответную систему с возможностью предварительной обработки вводимого текста. Одно из главных преимуществ использования векторизации — относительная простота реализации. На сегодняшний день существует большое множество сторонних библиотек машинного обучения для обработки и анализа информации на естественном языке.

Используя метод векторизации возможно создать большое множество различных узкоспециализированных вопросно-ответных систем.

### IV. СПИСОК ЛИТЕРАТУРЫ

1. Gigabaza.ru [Электронный ресурс]. – Режим доступа: <https://gigabaza.ru/doc/67598.html>. – Дата доступа: 21.10.2021.
2. Nlpub.ru [Электронный ресурс]. – Режим доступа: <https://nlpub.ru/Rumorhу>. – Дата доступа: 21.10.2021.
3. Хобсон, Л. Обработка естественного языка в действии / Л. Хобсон, Х. Ханнес, Х. Коул. – СПб // Питер. – 2020. – С.575

# ШАХМАТНЫЙ ДВИЖОК

Гудков А. С., Навроцкий А. А.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: gudkou\_fit u@mail.ru, navrotsky@bsuir.by

*Рассматривается алгоритм создания шахматного движка, позволяющий эффективно рассчитывать наилучшие ходы в шахматной партии.*

## ВВЕДЕНИЕ

В настоящее время становятся популярными компьютерные шахматные программы. Для реализации этих программ используются шахматные движки, которые позволяют на основе анализа шахматных позиций возвращать наиболее оптимальные варианты ходов.

### I. ПРИНЦИП РАБОТЫ ШАХМАТНОГО ДВИЖКА

Создание шахматной программы включает несколько этапов:

- визуализация доски;
- перемещение фигур;
- оценка доски;
- дерево поиска;
- Alpha-Beta;
- улучшенная функция оценки.

Первые два этапа реализуют оболочку программы (интерфейс, обеспечивающий взаимодействие пользователя с программой). Они легко осуществляются, например, с помощью JavaScript фреймворка Vue.js. Остальные этапы представляют собой алгоритм вычисления следующего хода (шахматный движок). Исследуя текущую позицию, движок просчитывает большое количество возможных вариантов шахматных ходов, численно оценивая каждый вариант и представляя наилучший результат пользователю. Огромное количество комбинаций, которые необходимо перебрать программе существенно замедляет работу движка, поэтому большой интерес представляют разработки критерием отсеивания заведомо невыигрышных шагов.

### II. ОЦЕНКА ДОСКИ

Общее количество уникальных партий в шахматы превышает количество атомов во Вселенной, отчего невозможно найти наилучший ход простым перебором возможных вариантов позиций. Поэтому в шахматных движках используются алгоритмы оценки позиций и деревьев поиска возможных ходов.

Для оценки расстановки силы на шахматной доске, в самом простом случае можно рассчитать относительную силу фигур, установив стоимость для каждой фигуры (пешка – 1, конь и офицер – 3, тура – 5, ферзь – 9, а король – 900, т. к. его утрата приводит к проигрышу партии).

Положим для белых цену фигур положительной, а для черных – отрицательной. Исходная матрица ходов представлена на рисунке 1.

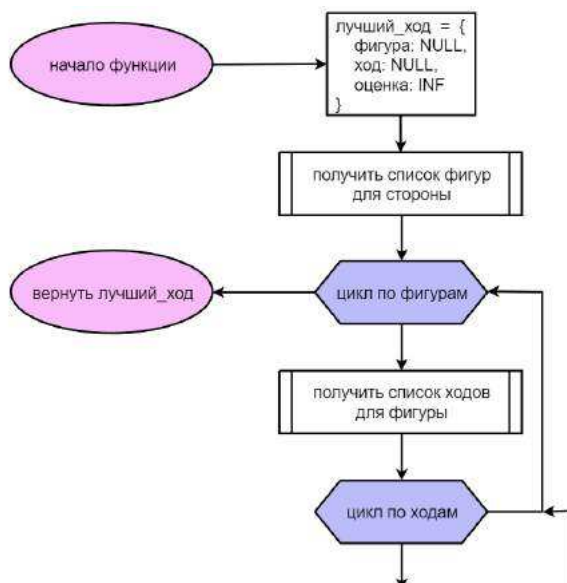
```
const BOARD = [
 [-5, -3, -4, -9, -900, -4, -3, -5],
 [-1, -1, -1, -1, -1, -1, -1, -1],
 [0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0],
 [0, 0, 0, 0, 0, 0, 0, 0],
 [1, 1, 1, 1, 1, 1, 1, 1],
 [5, 3, 4, 9, 900, 4, 3, 5]
];
```

Рис. 1 – Матрица ходов

Предложенная оценка доски позволяет выбирать ход с максимальной оценкой.

### III. ДЕРЕВО ПОИСКА

Поиск оптимального хода для выигрыша в шахматной партии требует расчета ситуации на несколько ходов вперед. Для этого создаётся дерево поиска, анализирующее все возможные ходы до заданной глубины. Полученная оценка возвращается в родительский узел. Блок-схема дерева поиска представлена на рисунке 2. В жёлтом блоке происходит вызов аналогичной рекурсивной функции поиска для создания новой ветви в дереве.





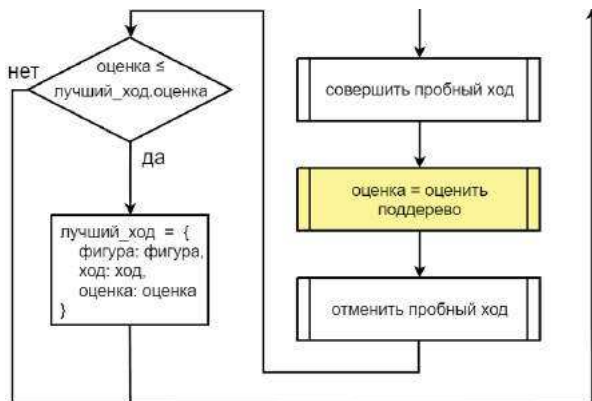


Рис. 2 – Блок-схема корня дерева поиска

С деревом поиска алгоритм начинает понимать базовую тактику шахмат и уже способен не только составить конкуренцию, но и обыграть большинство игроков.

#### IV. АЛФА-БЕТА

Стоит отметить, что при увеличении глубины поиска алгоритма повышается его эффективность, но также возрастает и время. Используя переменную-счётчик, было получено количество итераций для разработанного движка с различными уровнями глубины поиска. Результаты представлены на таблице 1.

Таблица 1 – зависимость количества итераций от глубины дерева

| Глубина дерева | Количество итераций | Время, с |
|----------------|---------------------|----------|
| 1              | 10-20               | 0.1-0.2  |
| 2              | 500-700             | 0.3-0.9  |
| 3              | 14000-98000         | 1.1-5.2  |
| 4              | 500000-4000000      | 5.9-8.7  |
| 5              | 6000000 и более     | 9.6-14.6 |

Для шахмат среднее количество возможных перемещений из одной позиции примерно равно 40. Следовательно, для расчета дерева с глубиной, равной 4, необходимо перебрать  $40 * 40 * 40 * 40 = 2560$  тыс. позиций, а на глубину 5 – 10240 тыс. Дерево поиска, растёт экспоненциально, поэтому для большой глубины дерева простой перебор использовать невозможно. На сегодняшний день на самых мощных процессорах при самом оптимальном коде можно рассчитать глубину дерева равную 6 в реально оцениваемый промежуток времени.

Так как при простом переборе алгоритм оценивает все возможные ходы партии, в том числе и бессмысленные, то следует предусмотреть возможность «отсечения» заведомо невыигрышных цепочек. Эля этого хорошо подходит алгоритм Alpha-Beta-отсечения, который является основой всех современных шахматных движков.

Для определения отсекаемых вариантов в рекурсивной функции требуется ввести две но-

вые переменные максимум для белых (ALPHA) и максимум для черных (BETA). При первом вызове обе эти величины равны минимально возможному значению (-INFINITY). Если найденная позиция, лучше текущей, то это значение запоминается. Ветвь, оценка которой не улучшает уже найденное оптимальное значение, будет «отсечена». Использование алгоритма Alpha-Beta позволяет значительно увеличить глубину поиска, затрачивая прежний объём ресурсов.

#### V. УЛУЧШЕННАЯ ФУНКЦИЯ ОЦЕНКИ

Качество работы алгоритма зависит от способа оценки относительной силы фигур, поэтому в работе предложена оценка, учитывающая кроме типа фигуры ещё и позицию фигуры на шахматной доске. Например, конь, размещенный в центре шахматной доски, более ценен, так как количество доступных ходов для этой фигуры увеличивается. Ценность пешки возрастает с приближением к противоположной стороне доски (при достижении противоположной стороны пешка становится ферзём). Маска ценности пешки на шахматной доске представлена на рисунке 3.

```
const BOARD_MASK_WHITE_PAWN = [
 [7.0, 7.0, 7.0, 7.0, 7.0, 7.0, 7.0, 7.0],
 [5.0, 5.0, 5.0, 5.0, 5.0, 5.0, 5.0, 5.0],
 [1.0, 1.0, 2.0, 3.0, 3.0, 2.0, 1.0, 1.0],
 [0.5, 0.5, 1.0, 2.5, 2.5, 1.0, 0.5, 0.5],
 [0.0, 0.0, 0.0, 2.0, 2.0, 0.0, 0.0, 0.0],
 [0.5, -0.5, -1.0, 0.0, 0.0, -1.0, -0.5, 0.5],
 [0.5, 1.0, 1.0, -2.0, -2.0, 1.0, 1.0, 0.5],
 [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
];
```

Рис. 3 – Маска ценности пешки в зависимости от её позиции на шахматной доске

Аналогичным образом, основываясь на шахматной логике, составляются маски ценностей для остальных фигур.

#### VI. ВЫВОДЫ

Представлен алгоритм, на основе которого разработан шахматный движок, позволяющий рассчитывать оптимальные дебюты в шахматной игре.

#### VII. СПИСОК ЛИТЕРАТУРЫ

1. Программирование шахмат и других логических игр / Е. Н. Корнилов. – СПб.: БХВ-Петербург, 2005. – 272 с.: ил.
2. Simple chess AI [Электронный ресурс]. – Режим доступа: <https://github.com/lhartikk/simple-chess-ai>. – Дата доступа: 14.03.2021.
3. Chess program [Электронный ресурс]. – Режим доступа: <https://github.com/danie/Chess>. – Дата доступа: 14.03.2021.
4. Разработка шахматной программы [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/329528>. – Дата доступа: 14.03.2021.

# ИСПОЛЬЗОВАНИЕ НЕЛИНЕЙНОГО СГЛАЖИВАНИЯ ДЛЯ АНАЛИЗА ДАННЫХ

Гринюк Д. А., Олиферович Н. М., Сухорукова И. Г., Карпович М. Д., Михновец А. С.  
Кафедра автоматизации производственных процессов и электротехники,  
Белорусский государственный технологический университет  
Минск, Республика Беларусь  
E-mail: hryniukda@gmail.com

*Рассмотрены варианты обработки данных с первичных измерительных преобразователей. Повышение чувствительности современных преобразователей сопряжено с высоким уровнем помех. Одновременно существенно растет вычислительная мощность микропроцессоров для использования более сложных алгоритмов фильтрации. В качестве объекта рассмотрен сигнал с датчика геометрических перемещений при измерении нелинейной деформации. Для задачи был выбран вариант кусочной аппроксимации квадратичной функцией. Оптимизация алгоритма сглаживания проводилась путем использования модельного сигнала подобного измеренному. Для повышения соотношения сигнал/шум предложено использовать двойное сглаживание.*

## I. ВВЕДЕНИЕ

При обработке трендов с первичных преобразователей всегда встает вопрос о рациональном уменьшении шумов. Этот вопрос решается достаточно просто для измерительных преобразователей с несущей первой гармоникой. Фильтры способны увеличить динамическую ошибку, но искажение информации происходит минимально. Однако при анализе трендов в научных или научно-производственных лабораториях нужно сохранить форму сигнала. Например, такие задачи стоят при обработке сигналов в тамографии.

Для линейных фильтров характерно внесение искажений за счет различного влияния на амплитуду и фазу гармоник полезного сигнала. Разнообразие методов, используемых в задачах анализа и моделирования временных рядов, велико [1]. При обработке данных используют методы математической статистики: критерий согласия Колмогорова, теорема Вальда, теорема Гюфдинга и т.д. Сейчас широко используются различные нечеткие, нейронные, вейвлет-преобразования и другие методы. В научных лабораториях могут различаться задачи по выделению полезной информации из измерительных трендов, поэтому происходит развитие методов обработки временных рядов. В литературе можно встретить варианты использования фильтров, основанные на скользящем взвешенном среднем; скользящем среднем арифметическом; методе Савицкого – Голея; сглаживании гауссианой и т.д.

Вариант многократного сглаживания обсуждается в ряде источников. Однако, согласно центральной предельной теореме теории вероятностей [2], выигрыш маловероятен. Это явление не зависит от метода сглаживания. При увеличении числа повторов  $n$  результат будет приближаться к однократному сглаживанию скользящим взвешенным средним с весами, распреде-

ленными по закону Гаусса. Однако многократное сглаживание имеет смысл при комбинации методов с различным принципом и при необходимости оптимизации вычислительной нагрузки на устройство.

## II. МЕТОДИКА ИССЛЕДОВАНИЯ

В процессе оптимизации обработки измерительных трендов механических деформаций композитных структур была выявлена проблема сильной зашумленности (рис. 1). Стояла задача выбора метода и настроек для максимального сохранения формы сигнала. Для обработки данных был выбран алгоритм с использованием метода наименьших квадратов [3, 4].

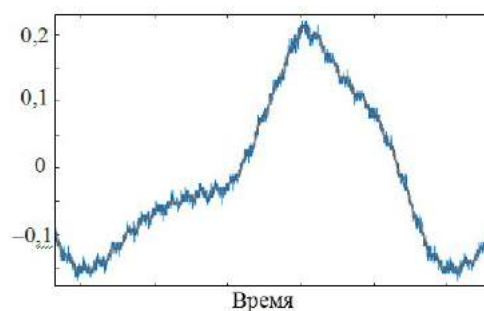


Рис. 1 – Зашумленный измерительный сигнал

Параметром настройки метода является ширина окна усреднения  $N$  и возможность использования многократного сглаживания. Для оценки их влияния на качество сглаживания произведено моделирование и последующий анализ двух сигналов с наложением шумов и помех, которые характерны для исходной задачи. Первый сигнал  $MS_1$  является гармоническим сигналом. Другой сигнал является суммой пяти гармоник, которые были получены из сигнала на рис.1. Каждый из модельных сигналов имел эквивалентное количество точек на период (8022), как и у экспериментального сигнала.

Модельные сигналы сглаживались алгоритмом дважды в разных комбинациях окон первичного и вторичного сглаживания  $N_1$  и  $N_2$ . Значения окон выбирались из ряда: 5, 9, 13, 17, 23, 29, 35, 43, 51, 105, 151, 281, 341, 381, 421, 521, 621, 721, 821, 921, 969, 1017, 1065, 1111, 1301, 1491. Для оценки качества, определялось значение мощности шумов и полезных сигналов с использованием спектрального анализа и теоремы Парсеваля.

### III. АНАЛИЗ РЕЗУЛЬТАТОВ

Эффективность подавления помех путем однократного и двукратного сглаживания можно видеть на рис.2–3.

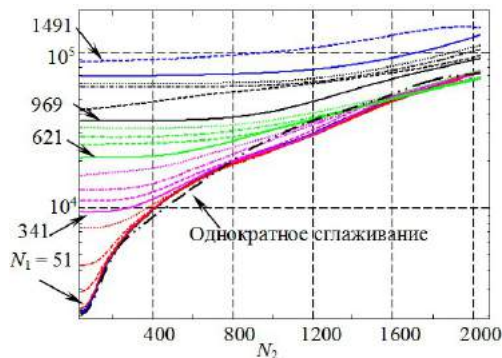


Рис. 2 – Отношение "исходный сигнал/шум" после сглаживания для  $MS_1$

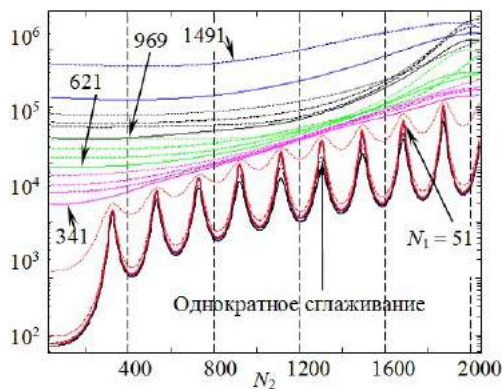


Рис. 3 – Отношение "исходный сигнал/шум" после сглаживания для  $MS_2$

Анализ результатов моделирования показывает, что после значения  $N_2$  1000–1200 для модельного сигнала  $MS_1$  намечается устойчивая тенденция изменения амплитуды основного сигнала. При этом, если первичное сглаживание происходило в диапазоне до 300, то амплитуда полезного сигнала незначительно изменяется. Оптимальным выбором здесь будет двукратное первичное  $N_1 = 250$ –300 и вторичное  $N_2 = 1200$ –1800 сглаживание. Даже при  $N_1 = 621$  и  $N_2 = 1200$ –1400 позиция более выигрышная, чем при однократном сглаживании.

В сглаживании модельного сигнала  $MS_2$  происходят практически такие же изменения отношения сигнал/шум в зависимости от  $N_1$  и  $N_2$ .

Однако наличие гармонической помехи показывает существенное влияние на результат сглаживания правильный выбор значения величины  $N_1$  и  $N_2$  как при однократном, так и при двукратном сглаживании.

В случае двукратного сглаживания при значениях  $N_1$  выше 341, гармоническая помеха не проявляется, что соответствует трем её периодам.

Нелинейный характер работы фильтра не всегда позволяет прогнозировать его работу. Современные цифровые системы преобразования редко формируют тренды с нормальным распределением шумов и помех. Кроме полезной информации, очень часто в измерительном тренде наблюдаются помехи от сети питания и выпрямителей, особенностей работы аналогово-цифровых преобразователей (рис. 1).

Полученные результаты оптимизации параметров двукратного сглаживания были использованы для обработки измерительных данных в эксперименте, что отмечалось в начале. После моделирования и были определены компромиссные значения  $N_1$  и  $N_2$ , при которых наблюдаются небольшие изменения мощности полезных гармоник и наибольшее подавление шумовых составляющих. Для измерительного тренда было характерно плавание частоты полезного сигнала. Точная частота сигнала определялась путем вариации частоты при разложении в ряд Фурье и нахождения момента максимума амплитуды первой гармоники. После оптимизации процесса сглаживания флуктуации частоты не оказывали влияние на качество обработки измерительной информации.

### IV. ВЫВОДЫ

Использование метода локальной аппроксимации по средствам наименьших квадратов предоставляет гибкие возможности по сглаживанию измерительной информации. Нелинейный характер преобразования позволяет получить эффект от многократного сглаживания на практике, несмотря на отсутствие теоретических предпосылок. Имитационное моделирование измерительных трендов позволяет оптимизировать параметры сглаживания.

1. Hangos, K. M. Process modelling and model analysis. /K. M. Hangos, I. T. Cameron – San Diego : Academic Press, 2001. – 543 p.
2. Вентцель Е. С. Теория вероятностей. – М.: Высшая школа, 1999. – 576 с.
3. Гринюк, Д. А. Использование алгоритмов аппроксимации для сглаживания трендов измерительных преобразователей / Гринюк Д. А., Сухорукова И. Г., Олиферович Н. М. // Труды ВГТУ. Сер. 3, Физико-математические науки и информатика. 2017. – № 2 (200). – С. 82–87.
4. Hryniuk, D. Adaptive smoothing and filtering in transducers / D. Hryniuk, I. Suhorukova, M. Olyferovich // 2016 Open Conference of Electrical, Electronic and Information Sciences (eStream 2016), Vilnius, Lithuania, 21 April 2016. 2016. p. 1–4.

# УЛУЧШЕНИЕ НАДЕЖНОСТИ ПРОГРАММНЫХ СРЕДСТВ С ПОМОЩЬЮ ПРИМЕНЕНИЯ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ

Денисик П. Д.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: fd1crash@gmail.com

*В данной работе рассматриваются изменения в области тестирования и улучшение надежности программных средств при помощи использования современного инструмента такого как автоматизированное тестирование.*

## ВВЕДЕНИЕ

С каждым годом разрабатывается все больше различных программных средств. Быстрота и качество разработки программных средств выходят на первый план. Одним из главных этапов разработки является оценка надежности программного средства с помощью применения комплексных технологий тестирования. Одной из таких технологий является – автоматизированное тестирование.

Автоматизация тестирования помогает сократить время на проверку качества прикладного программного средства и подходит для проведения множества регулярных операций – от имитации работы пользователя до создания отчетов о состоянии тестирования. Кроме того, сервис позволяет проводить проверки, недоступные при ручном тестировании. Автоматические тесты разрабатываются индивидуально, с учетом всех характеристик программного продукта.

## I. ИЗМЕНЕНИЯ В ОБЛАСТИ ТЕСТИРОВАНИЯ

Изменения в области тестирования обусловлены как внешними, так и внутренними факторами. Давайте рассмотрим некоторые из них.

Внешние факторы:

- Широкое использование технологий в повседневной жизни. Согласно отчету компании, App Annie за 2017 год, мы используем в среднем девять приложений в день, и это только на смартфонах. Подумайте, как измениться это число, если учесть остальные технологии, которые вы используете в течение дня.
- повышение ожиданий пользователей от программного обеспечения. Например, TechBeason выяснил, что 49 процентов пользователей ожидают, что приложения отреагируют на запрос за две секунды или меньше, и 80 процентов пользователей не будут использовать приложение с ошибками более трех раз.
- потребность в разработке нового и улучшениях существующего программного обеспе-

чения. Мир не стоит на месте, технологии постоянно развиваются.

Согласно исследованию IDC, к 2020 году мировой рынок продуктов и услуг Internet of Things достиг 7,1 триллиона долларов, что создало необходимость для каждой компании улучшать разработку и тестирование ПО.[1]

Внутренние факторы:

- Рост технологических решений, помогающих на протяжении жизненного цикла разработки программного обеспечения (SDLC). Согласно графику, SDLC 2017 года, существует чуть более 1500 различных инструментов, которые тестировщики могут использовать в рамках SDLC, но неправильное их использование может значительно усложнить работу.
- рост Agile и DevOps ошутимо влияет на тестирование программного обеспечения. По данным VersionOne, 95 процентов компаний используют Agile, а по данным RightScale, внедрение DevOps достигло 74 процентов. Таким образом, тестировщикам необходимо постоянно адаптироваться.
- Автоматизация быстро становится подходом по умолчанию, однако тестировщики должны быть осторожны в поиске правильного баланса между автоматическим и ручным тестированием.

Брэндон Сайпс, вице-президент DevOps в CPrime прогнозирует, что «будущее тестирования программного обеспечения — это почти полностью автоматизированные тестовые кейсы. Автоматизация — это замечательно, но, если вы автоматизируете не ту область, вы не оберетесь проблем», — добавляет он.

## II. ПРИМЕНЕНИЯ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ

Примеры, когда стоит применять автоматизированное тестирование представлены ниже [2]:

- Ваша первоочередная задача – сэкономить время проектной команды;
- тесты должны выполняться для каждой сборки приложения;

- ваш проект длительный или комплексный (состоит из различных итераций);
- на выполнение тест-кейсов тратится много времени и ресурсов;
- проводится нагрузочное или стресс-тестирование;
- нужно сократить текущий объем тестирования с целью успеть к определенным срокам.

При построении грамотного процесса по внедрению автоматизированное тестирование может быть очень прибыльным, но у этого вида тестирования есть свои недостатки:

- Внедрение автоматизации – это не однократное действие, так как нужно тратить время на поддержку актуальности автотестов;
- количество новых регрессионных дефектов всегда будет немного меньше по сравнению с теми ошибками, что будут обнаружены в новой функциональности;
- иногда результаты тестирования нужны в течение нескольких часов, особенно если говорить о проверках недавно внедренной функциональности. В этом случае эффективнее перейти к ручному тестированию и получить результаты как можно скорее.

Но плюсы от применения автотестов для проекта могут перекрывать любые минусы. Самые весомые плюсы:

- Быстрая обратная связь о качестве приложения;
- раннее обнаружение дефектов программного обеспечения;
- экономия времени;
- безопасность тестовых данных;
- быстрая скорость выполнения проверок.

Преимущества внедрения автоматизированного тестирования показано на рисунке 1.

Из представленного на рисунке можно сделать следующие выводы:

1. Правильное внедрение автоматизированного тестирования помогает заметно улучшить качество программного средства за счет более полного тестового покрытия, а также позволяет обнаружить дефекты на более ранних этапах разработки.
2. автоматизированное тестирование делает гораздо более прозрачным и контролируемым процесс тестирования программного средства.
3. автоматизированное тестирование позволяет сильно уменьшить время затрачиваемого для тестирования, а значит помогает экономить денежные и человеческие ресурсы компании.
4. с применением автоматизированного тестирования значительно сокращаются общие затраты на тестирование.

### ЗАКЛЮЧЕНИЕ

На основе предоставленных данных можно сделать выводы: внедрять автоматизированное тестирование можно только там, где это необходимо, а также создавать такую систему автоматизированного тестирования, которую смогут легко поддерживать и расширять разработчики. Автоматизированное тестирование очень эффективно для проверки обратной совместимости. Также стоит отметить грамотное сочетание ручного тестирования и автоматизированного, для получения максимальной эффективности связки двух подходов.

### СПИСОК ЛИТЕРАТУРЫ

1. Trends Shaping the Future of Software Testing [Electronic resource] / Tricentis Corp. – Mode of access: <https://www.tricentis.com/blog/5-trends-future-software-testing>. – Date of access: 14.08.2017.
2. Автоматизация тестирования от «А» до «Я»: часть 1 [Электронный ресурс] / A1QA Corp. – Режим доступа: <https://www.a1qa.ru/blog/avtomatizatsiya-testirovaniya-ot-a-do-ya-chast-1>. – Дата доступа: 15.07.2019.

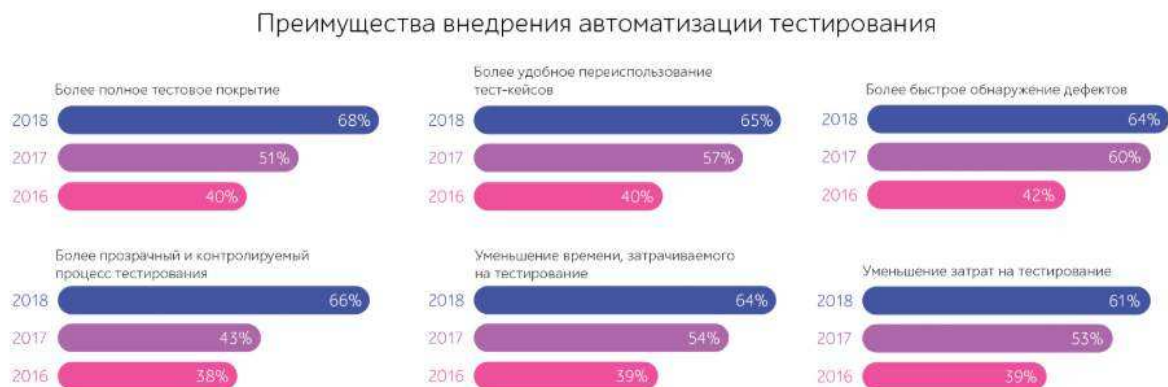


Рис. 1 – Преимущества внедрения автоматизации тестирования

# АЛГОРИТМ ОПИСАНИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ НЕЗАВИСИМОГО ИГРОВОГО МИРА

Дмитриев А. С.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: lex150498@mail.ru

*Рассматривается алгоритм и принципы описания искусственного интеллекта для использования в игровых мирах компьютерных игр независимого типа для усиления эффекта погружения игрока в игровой процесс.*

## ВВЕДЕНИЕ

Игровой мир компьютерной игры может быть зависимым и независимым от действий игрока. В случае независимого мира большая часть процессов в нём протекает без внешнего вмешательства. Задачей такого мира является демонстрация игроку своей полноценности путём имитации процессов реального мира. Поведение игровых персонажей в таких мирах является одним из важнейших элементов, так как обычно с ними часто взаимодействует игрок.

## I. АКТУАЛЬНОСТЬ

Персонажи под управлением искусственного интеллекта в играх могут вести себя абсолютно по разному, но в мирах с имитацией реальных процессов их основная задача - это создать у игрока ощущение взаимодействия с чем-то живым и самобытным. Данный алгоритм даёт возможность сбалансировать между собой затраты по производительности и интересность искусственного интеллекта для игрока.

## II. ОСНОВНЫЕ ФАКТОРЫ УЧИТЫВАЕМЫЕ В АЛГОРИТМЕ

При описании алгоритма основными факторами стали:

1. Самодостаточность – у игрового персонажа под управлением искусственного интеллекта всегда должно быть занятие. Даже если он ожидает некоего события, то это должно быть похоже на ожидание живого существа, например, проигрывать анимация ожидания, с, к примеру, покачиваниями тела, изменением направления взгляда и так далее. Персонаж ни в коем случае не должен выглядеть как бездушная статуя.
2. Производительность – так как персонажи всегда должны быть заняты неким делом, то их действия будут нагружать вычислениями центральный процессор, поэтому вычисления должны быть минимизированы, особенно если в игровом мире действует одновременно большое количество персонажей. Иногда для оптимизации стоит пожертвовать функционалом, так как ни

один игрок не будет рад какой-нибудь игровой детали, если при этом игра работает нестабильно.

3. Изменчивость – шаблонное поведение персонажей очень быстро приедается игроку. Это не значит, что искусственный интеллект должен постоянно удивлять игрока чем-то новым, но небольшие изменения маршрутов, действий или реакций на события хорошо разбивают шаблонность и делают персонажей менее предсказуемыми, а значит более интересными для игрока.

## III. ОСНОВНЫЕ ПРИНЦИПЫ АЛГОРИТМА

При описании алгоритма были выработаны следующие принципы его построения:

1. Основная логика персонажей должна быть представлена в виде деревьев поведения. Такой подход требует описания большого количества ситуаций, но он будет куда более гибким и производительным, чем построение логики на основе нейронных сетей, так как они хорошо работают с узкоспециализированными задачами, но хуже с широкопрофильными, как, например, с имитацией поведения. При этом отдельные сегменты побочной узконаправленной логики, к примеру поиск оптимального укрытия или пути, можно сделать с использованием нейронной сети, если при этом не страдает производительность.
2. У персонажей должны быть свои области влияния: своё расписание действий, свои территории передвижения и активности, свои социальные группы, и своё положение в иерархии. Например, персонаж с более высоким приоритетом в иерархии может передать выполнение одной из задач из своего расписания в расписание персонажа с более низким приоритетом, но с учётом их нахождения в одной социальной группе.
3. Персонажи из взаимодействующих групп должны без ведома игрока обмениваться вычислительной информацией в целях оптимизации и быстрого действия. К примеру если один из персонажей атакующий, а второй должен от него искать укрытие, то ата-

кующий должен посчитать с учётом своих параметров эффективность атаки ближайших к обороняющемуся укрытий и передать эту информацию ему. При получении такой информации от нескольких противников обороняющийся сможет узнать суммарную эффективность каждого из укрытий и уже рассчитывать её с остальными параметрами, например дистанцией до укрытия. При этом когда атакующий обнаруживает обороняющегося, то он сразу же проводит данные расчёты и передаёт их обороняющемуся одновременно и передавая информацию, и уведомляя его, что он обнаружен. Это избавляет от лишних проверок для обороняющегося был ли он замечен или нет и даёт ему оперативную информацию по факту обнаружения.

#### IV. АЛГОРИТМ ОПИСАНИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

В ходе работы над алгоритмом были выработаны следующие шаги разработки искусственного интеллекта:

1. Описать логику пространств в которых может перемещаться искусственный интеллект.
2. Задать формальные правила создания уровней игрового мира, основываясь на логике пространств.
3. Описать логику перемещения и поиска путей искусственного интеллекта в подготовленных для него пространствах.
4. Задать ключи для изменения состояний искусственного интеллекта.
5. Создать счётчик времени на уровне и основываясь на нём описать функционал для создания расписаний активностей искусственного интеллекта.
6. Создать зоны активности персонажей.
7. Наполнить зоны активности предметами для взаимодействия.
8. Описать типовое поведение искусственного интеллекта по расписанию в зонах активности.
9. Присвоить персонажам группы и описать взаимодействие в них.
10. Создать иерархию в группах и между группами.
11. Добавить возможность искусственному интеллекту передавать задачи вышестоящим в иерархии единиц нижестоящим.
12. Присвоить зонам активности статусы и условия доступности для различных групп.
13. Описать генератор расписаний с учётом параметров персонажей и их групп.
14. Описать логику взаимодействия групп между друг другом.
15. Описать поведение искусственного интеллекта в нестандартных для единиц и групп ситуациях.

Далее описаны некоторые решения для шагов алгоритма. В качестве границ зон действий было решено выбрать все замкнутые или частично замкнутые контуры в игровом мире, состоящие из объектов, через которые не могут проходить игровые персонажи, а в качестве переходов установить триггеры на местах границ где персонажи могут проходить, чтобы уведомлять их о смене зоны. Зонам были присвоены идентификаторы доступности для разных групп персонажей, периоды времени данной доступности и выполняемые внутри активности. Персонажам были присвоены социальные группы, расписание возможных активностей и черты характера, представленные числовыми значениями, и действующие как ключи при выборе последующих действий в дереве поведения. При создании логики передачи задач в расписании между персонажами учитывалась не только позиция в иерархии, но также может ли получатель задачи её выполнять и есть ли у него доступ в необходимую зону. Если доступа нет, то он выдаётся, а в расписании передавшего задачу на освобождённом месте создаётся новая активность. После описания типовых ситуаций поведения для персонажей было описано нетиповое поведение с реакциями на игрока и других персонажей. К примеру, если персонаж при неких обстоятельствах, например, нанесении ему урона, получает статус испуга и принимает решение бежать, то он начинает игнорировать правила доступа в зоны, и может забежать туда, где при типовых условиях находиться не должен.

#### V. ВЫВОДЫ

В ходе тестов выяснилось что алгоритм работает стабильно с большим количеством персонажей, но с условием равномерного переключения их между активностями, чтобы не создавать большое количество одновременных изменений состояний персонажей и обчёта путей передвижения.

#### VI. ЗАКЛЮЧЕНИЕ

Данный алгоритм поведения имеет высокий коэффициент масштабируемости, но при его использовании стоит установить границы возможного взаимодействия игрока с ним, так как лучше сделать некое законченное поведение с необходимыми вариациями, чем описывать множество ситуаций с которыми игрок возможно даже ни разу не столкнётся.

#### VII. СПИСОК ЛИТЕРАТУРЫ

1. Artificial Intelligence: A Modern Approach / J. R. Stuart, P. Norvig // Pearson, 1994.
2. AI Game Programming Wisdom / S. Rabin // Cengage Learning, 2002.
3. AI for Game Developers / D. M. Bourg, G. Seemann // O'Reilly Associates, 2004.

# ТОЧНОЕ ЗНАЧЕНИЕ ВЕРОЯТНОСТИ ПРОПУСКА ОШИБКИ ПРИ НАБЛЮДЕНИИ ВЕКТОРОВ ПЕРЕХОДОВ В АСИМПТОТИКЕ

Кобяк И.П.

Кафедра электронных вычислительных машин,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: ipkobyak2012@mail.ru

*В представляемой работе на основе принципа включения и исключения получены соотношения, позволяющие определить вероятность пропуска ошибки при наблюдении векторов переходов в последовательностях со случайной природой. Рассмотрен частный случай производящей функции для одной переменной, характеризующей практически точное значение вероятности ошибки в асимптотике. Показано, что метод наблюдения векторов переходов является наиболее эффективным инструментом идентификации последовательностей по отношению к общеизвестным алгоритмам свертки.*

## ВВЕДЕНИЕ

Формирование векторов переходов (ВП) заданного вида осуществляется путем технического дифференцирования статических векторов случайного сообщения с последующим «однополупериодным сглаживанием» отрицательных событий. Достоинством принципа наблюдения таких субдинамических объектов является низкая вероятность неперехода сформированной оценки в некоторый другой класс эквивалентностей по отношению к классу нулевой гипотезы. При этом под классом нулевой гипотезы понимают количество конечных последовательностей длиной  $n$  заданного случайного процесса, дающих одинаковый контрольный код.

Возможность формирования факториальных моментов на основе эnumераторов специального вида позволяет определить ряд параметров характеризующих свойства точечных оценок, интересующих пользователей ряда прикладных задач. В частности, особый интерес вызывает принцип вычисления нормированной площади под кривой графика распределения вероятностей в асимптотике. Это позволяет выполнить сравнение показателей предлагаемого метода с другими известными алгоритмами свертки.

### 1. ПРОИЗВОДЯЩАЯ ФУНКЦИЯ ДЛЯ ВЕРОЯТНОСТИ НАБЛЮДЕНИЯ ВП

Пусть  $p$  — это теоретическая вероятность регистрации ВП заданного вида. Тогда основной результат данной работы может быть сформулирован в виде теоремы.

Теорема. Эnumератор вида

$$m_K(0) = \sum_g^{n-1} \left( px_1^1 + \sum_{i=2}^{n-2} px_1^i \frac{1}{2^{i+1}} \beta_i + \frac{1}{pm} \right)^{n-g}, \quad (1)$$

$$g = 2j\xi + 3j\psi + \dots + (n-1)j\eta$$

является производящей функцией для точного значения вероятности наблюдения ВП заданного вида, где

$$\beta_i = \frac{1}{\sqrt{1-4p}} \left[ (1 + \sqrt{1-4p})^{i+1} - (1 - \sqrt{1-4p})^{i+1} \right],$$

а целочисленные значения  $\xi, \psi, \dots, \eta$  — характеризуют наличие или отсутствие в наблюдаемой последовательности сложных объектов с длиной постобъектов равной  $i$ .

Доказательство. Сформируем сложные объекты, состоящие из композиции динамического вектора перехода с вероятностью [1]:

$$p = \frac{3^{r-\mu}}{m^2},$$

и постобъекта длиной  $i$ . Тогда, учитывая все разбиения многомерной выборки вида

$$n = 3j\xi + 4j\psi + \dots + nj\eta, \quad j = 1,$$

можем записать соотношение для вероятности пропуска ошибки при регистрации векторов переходов (или субдинамических векторов) в последовательности длиной  $n$ :

$$m_K(0) = \sum_g^{n-1} \frac{1}{m^{n-\sum_{i=1}^{n-2} ik_{1,i}}} \sum_{n-g} (3^{r-\mu})^{k_{1,1}} \times$$

$$\times \prod_{i=2}^{n-2} \left( \frac{3^{r-\mu}}{2^{i+1}} \beta_i \right)^{k_{1,i}} \frac{\left( \sum_{i=1}^{n-2} k_{1,i} \right)!}{\prod_{i=1}^{n-2} k_{1,i}!}, \quad (2)$$

при этом параметр  $g = 2j\xi + 3j\psi + (n-1)j\eta$  или  $g = 2jk_{1,1} + 3jk_{1,2} + (n-1)jk_{1,n-2}$ , где  $k_{1,i}$  — число сложных объектов с одним вектором перехода в последовательной серии из нескольких ВП.

Выполним преобразование факториальных моментов в соотношении (2), в результате чего получим функцию:

$$m_K(0) = \sum_g^{n-1} \sum_{n-g} \frac{1}{m^{n-g-\sum_{i=1}^{n-2} k_{1,i}}} \frac{1}{m^{\sum_{i=1}^{n-2} k_{1,i}}} \times$$

$$\times \left( \frac{3^{r-\mu}}{2^0} \right)^{k_{1,1}} \left( \frac{3^{r-\mu}}{2^{2+1}} \right)^{k_{1,2}} \beta_2^{k_{1,2}} \left( \frac{3^{r-\mu}}{2^{3+1}} \right)^{k_{1,3}} \beta_3^{k_{1,3}} \times$$

$$\times \dots \times \left( \frac{3^{r-\mu}}{2^{(n-2)+1}} \right)^{k_{1,n-2}} \beta_{n-2}^{k_{1,n-2}} \times$$



$$\times \frac{(k_{1,1} + k_{1,2} + k_{1,3} + \dots + k_{1,n-2})!}{k_{1,1}!k_{1,2}!k_{1,3}!\dots k_{1,n-2}!}.$$

Приведем подобные в степени числа  $m$  и сгруппируем члены с одинаковыми показателями. При этом имеем:

$$\begin{aligned} m_K(0) &= \sum_g \sum_{n-g} \frac{1}{m^{n-g-\sum_{i=1}^{n-2} k_{1,i}}} \left(\frac{3^{r-\mu}}{m^2}\right)^{k_{1,1}} \times \\ &\times \left(\frac{3^{r-\mu}}{m^2} \frac{\beta_2}{2^{2+1}}\right)^{k_{1,2}} \left(\frac{3^{r-\mu}}{m^2} \frac{\beta_3}{2^{3+1}}\right)^{k_{1,3}} \times \dots \times \\ &\times \left(\frac{3^{r-\mu}}{m^2} \frac{1}{2^{(n-2)+1}} \beta_{n-2}\right)^{k_{1,n-2}} \times \\ &\times \frac{(k_{1,1} + k_{1,2} + k_{1,3} + \dots + k_{1,n-2})!}{k_{1,1}!k_{1,2}!k_{1,3}!\dots k_{1,n-2}!}. \end{aligned}$$

Тогда, используя методику преобразования произведения в полиномиальную функцию, для произвольного момента времени  $t$  и параметра  $x_1$  можем записать:

$$\begin{aligned} m_K(t) &= \sum_g \left(x_1 p e^t + \frac{1}{2^{2+1}} x_1^2 \beta_2 p e^t + \dots + \right. \\ &\left. + \frac{1}{2^{(n-2)+1}} x_1^{n-2} \beta_{n-2} p e^t + \frac{1}{m}\right)^{n-g}. \quad (3) \end{aligned}$$

Откуда при  $t = 0$  и соответствующей полиномиальным коэффициентам аппроксимации  $\beta_i$ , получаем результат (1). Теорема доказана.

В соотношении (3) степень параметра равная  $i$  указывает на длину постобъекта, а произведение на вариативность частной функции правдоподобия в пределах данного постобъекта.

## II. ЧИСЛЕННОЕ ЗНАЧЕНИЕ ВЕРОЯТНОСТИ ОШИБКИ В АСИМПТОТИКЕ

Пусть случайный процесс характеризуют параметры  $r = 2$ ,  $\mu = 1$ , что определяет максимальную вероятность наблюдения ВП равную:

$$p_{\max} = \frac{3^{r-\mu}}{m^2} = \frac{3}{16}.$$

Тогда в соответствии с формулой (1) получаем:

$$\frac{1}{2^{i+1}} \beta_i = 2 \left[ (0,75)^{i+1} - (0,25)^{i+1} \right].$$

В асимптотике длина постобъекта  $i$  может быть рассчитана по формуле:

$$i = \frac{1 - 2p}{p} = \frac{10}{3}.$$

Соответственно значение  $g$  будет равно:

$$g = \frac{1 - p}{p} = \frac{13}{3}.$$

Тогда из сформированного выше эnumератора имеем упрощенное равенство вида:

$$\begin{aligned} m_K(0) &= \left[ \frac{3}{8} \left[ (0,75)^{i+1} - (0,25)^{i+1} \right] + \frac{1}{4} \right]^{n-g} = \\ &= (0,35688)^{n-4,333\dots}. \quad (4) \end{aligned}$$

Учитывая, что дробь в степени  $n$  (то есть равной длине выборки) есть бесконечно малая

величина можем констатировать факт весьма низкой вероятности пропуска ошибки при идентификации выборки контрольным кодом, соответствующим числу векторов переходов.

Используя логику формирования монообъектов с вероятностью указанной выше в доказанной теореме, можем записать также соотношение для вероятности ошибки в виде следующего равенства:

$$\begin{aligned} m_K(t) &\approx \frac{1}{m^n} (3^{r-\mu} m)^{np} \left[ \frac{1}{2^{i+1}} \frac{1}{\sqrt{1-4p}} \times \right. \\ &\times \left. \left[ (1 + \sqrt{1-4p})^{i+1} - (1 - \sqrt{1-4p})^{i+1} \right] \right]^{np} = \\ &= (0,3585228)^n. \quad (5) \end{aligned}$$

Полученные результаты незначительно отличаются по уровню мощности классов эквивалентностей, что следует из дробного представления параметров  $g$  и  $i$ , имеющих фактически статус целочисленных значений в общей форме функции (3).

## III. АСИМПТОТИКА ДЛЯ $r = \mu$

В более общем случае, то есть для вероятности регистрируемого события равной:

$$p = \frac{3^{r-\mu}}{m^2}, \quad r > 2,$$

можем записать другой крайний случай вероятности при равенстве параметров  $r = \mu$ :

$$p = \frac{1}{m^2} = \frac{1}{16}.$$

Тогда имеем:

$$i = m^2 - 2, \quad g = m^2 - 1.$$

С учетом данных значений и  $m = 4$  имеем:

$$\begin{aligned} m_K(0) &= \left( \frac{1}{16} 0,408115 + \frac{1}{4} \right)^{n-15} = \\ &= (0,275507)^{n-15}. \quad (6) \end{aligned}$$

Положим теперь значение  $r = 5$ , тогда:

$$\begin{aligned} m_K(0) &\approx \left[ \frac{1}{1022} \left( \frac{1}{2} + \frac{1}{2} \sqrt{\frac{32^2-4}{32^2}} \right)^{1023} + \frac{1}{32} \right]^{n-1023} = \\ &= (0,0316098)^{n-1023}. \quad (7) \end{aligned}$$

Сравнение результатов (6) и (7) показывает, что уровень вероятности ошибки с ростом разрядности наблюдаемого случайного процесса уменьшается:

$$\ln \frac{(0,0316098)^{n-1023}}{(0,275507)^{n-15}} \approx 2,676825$$

при  $n = 1\,000\,000$ .

- И.П.Кобяк. Асимптотика для вероятности пропуска ошибки при наблюдении векторов переходов // В сб. науч. статей VII Международной науч.-практич. конференции «BIG DATA and Advanced Analytics», Минск, 19-20 мая 2021 г. – Минск : Бестпринт, 2021. – С. 328–335.

# МОДЕРНИЗАЦИЯ АЛГОРИТМА ПОСТАВКИ ОБНОВЛЕНИЙ ДЛЯ СИСТЕМ МЕЙНФРЕЙМА

Колесников В. Г.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: leksilonchikk@gmail.com

*В данной работе рассматривается модернизация устаревшего метода построения дерева зависимостей единиц обновлений и их установки в системах мейнфреймов с применением современных технологий и минимизацией участия заказчика и посредников в процессе обновления.*

## ВВЕДЕНИЕ

Актуальность использования мейнфрейма как крупной системы по быстрой обработке данных является спорным вопросом на протяжении нескольких десятилетий. Еще недавно мейнфрейм претендовал на технологию, которой предрекали скорое забвение. Однако по сей день ведется разработка программ и модернизация существующих компонентов для этого комплекса подсистем, а отказ от его использования компанией скорее всего тесно связан с ее надвигающимся крахом.

Действующий алгоритм поставки обновлений для внутренних систем мейнфреймов включает в себя большое количество шагов, выполняемых вручную, с привлечением нескольких структурных единиц, производящих необходимые действия в определенных фазах. Следовательно, данный процесс требует максимальной автоматизации с минимизацией участия в ней инициатора и посредников.

### I. Действующий алгоритм поставки обновлений

Реализованная система со временем требует доработки существующих функций, а также обновления ее компонент с целью устранения ошибок или неточностей их работы. Изначальная реализация построения дерева зависимостей и поставки обновлений подразумевала минимальную автоматизацию последовательных действий данного процесса, так как основные задачи были не до конца обозначены, что усложняло реализацию полностью автоматизированной системы. Следовательно, созданный процесс исключал ручное вмешательство только в некоторых постоянно повторяющихся действиях. Главное преимущество данного процесса – предоставление единого сервиса для многих вариантов обновления как системы в целом, так и продуктовых систем мейнфрейма. [1–2]

Действующий механизм работает слаженно, но требует гораздо больше затрат на его реализацию чем его улучшенный вариант. Также, внедрение каких-либо изменений в уже существующий алгоритм требует слишком больших

затрат по времени и ресурсам разработчиков, как на этапе анализа проблем, так и при внесении этих самых изменений. Кроме того, данный метод подразумевает довольно сильную зависимость заказчика от процесса. Первое, на что необходимо обратить внимание – большое количество прерываний в процессе выполнения с целью введения дополнительных данных заказчиком или посредниками – теми членами процесса, которые знают систему непосредственно и имеют доступ ко внутренним данным со стороны разработчиков.

Следующая проблема данного подхода – большое количество промежуточных шагов для подготовки запускаемых программ. Этот изъян прежде всего связан с тем, что большая часть процессов находится и выполняется прямо на мейнфрейме. В те времена, когда данный подход реализовывался, не было необходимых возможностей для выноса некоторых шагов во внешнюю среду. Например, информирование заказчика об определенных шагах или действиях было возможно только через электронную почту либо непосредственно через мейнфрейм, однако в этом случае информация доставлялась только при непрерывной работе алгоритма, что исключало любые другие действия в мейнфрейме. Данный факт сильно затруднял развитие подхода в плане интерактивности.

На рис. 1 представлена простейшая диаграмма выполняемых действий для данного подхода.

### II. Модернизированный алгоритм поставки обновлений

Для устранения сразу нескольких архитектурных проблем действующего алгоритма поставки обновлений было решено в корне переработать сразу несколько шагов в процессе. Главная цель при этом – свести к минимуму вмешательство участника процесса, который выполняет запрос на обновление систем. Так, удалось преобразовать алгоритм к двум простым шагам для заказчика: предоставление данных и получение результата. При этом, первый шаг также

обладает возможностью автоматизации при согласии заказчика.

Кроме минимизации участия заказчика в процессе поставки обновлений также было необходимо устранить влияние каких-либо посредников на процесс. Поэтому, основные действия, ранее выполняемые агентами, теперь сведены к работе нескольких утилит, строящих последовательность действий автоматически на основе предоставляемых параметров.

Благодаря внедрению в процесс современных технологий, которые работают вне мейнфрейма, предполагалось урезание количества шагов для выполнения построения дерева зависимостей и поставки обновлений на его основе. То есть теперь те действия, которые требовалось выполнять исключительно на стороне мейнфрейма, производятся вне системы. Таким образом, то, что ранее производилось внутри системы заказчика, теперь позволяют разгрузить ресурсы мейнфрейма для выполнения им более специфических задач, не связанных с его обновлением. Данный факт может показаться ослаблением безопасности, однако эта проблема решается минимизацией взаимодействия внутренних компонентов со внешней средой и защитой всех входных и выходных каналов.

В качестве модернизации построения дерева зависимостей, выступает несколько утилит для сбора и анализа данных. Сейчас данная схема предоставляется в текстовом виде. В конечном результате планируется выдавать само дерево в интерактивном виде с построением в реальном времени.

Немаловажной проблемой устаревшего алгоритма является скорость предоставления данных о производимых процедурах. Модернизированный

подход позволяет отслеживать выполняемые действия в режиме реального времени, что облегчает анализ конкретных результатов поэтапно. Данная особенность дает возможность устранять проблемы на определенном этапе гораздо быстрее, чем это было при устаревшем алгоритме.

На рис. 2 представлена простейшая диаграмма выполняемых действий для улучшенного алгоритма.

## ЗАКЛЮЧЕНИЕ

На основе предоставленных данных можно сделать вывод, что модернизация алгоритма позволяет избежать некоторых промежуточных шагов в этом непростом процессе, что дает существенное ускорение по сравнению с устаревшим аналогом. Применение современных технологий не только облегчает и удешевляет процесс, но и позволяет ускорить процесс интеграции мейнфрейма в современный мир разработки и использования программного обеспечения. Все проблемы с безопасностью данного подхода решаются ограничением количества внешних каналов, а также строгой процедурой их введения в эксплуатацию.

## СПИСОК ЛИТЕРАТУРЫ

1. IBM CustomPac services [Electronic resource] / IBM Corp. – United States, 2010. – Mode of access: <https://www.ibm.com/services/ca/en/custompac/>. – Date of access: 17.10.2021.
2. Methods of installing z/OS products [Electronic resource] / IBM Corp. – United States, 2010. – Mode of access: <https://www.ibm.com/docs/en/zos/2.4.0?topic=zos-methods-installing-products>. – Date of access: 17.10.2021.

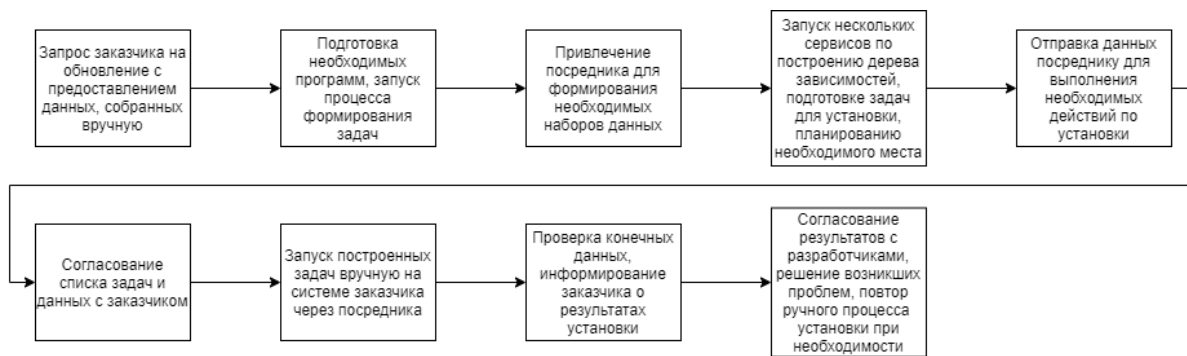


Рис. 1 – Действующий алгоритм поставки обновлений

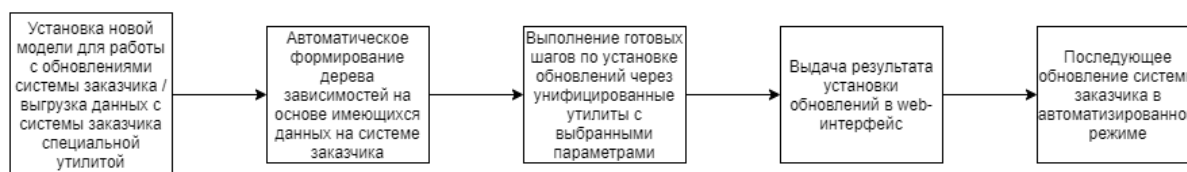


Рис. 2 – Модернизированный алгоритм поставки обновлений

# ПРОГРАММНЫЙ КОМПЛЕКС КОНТРОЛЯ РАБОТЫ СЕРДЕЧНО-СОСУДИСТОЙ СИСТЕМЫ ЧЕЛОВЕКА

Кожемякин П. Н., Парамонов А. И.  
Факультет компьютерных систем и сетей,  
кафедра программного обеспечения информационных технологий,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: bahmelki@gmail.com, antonparamonov@tut.by

*Проект выполнен в два этапа: аппаратный этап реализован на основе микроконтроллера Arduino для получения и предварительной обработки информации. Программный этап включает в себя разработку программного обеспечения для мобильного устройства на платформе Android. Программное средство предназначено для анализа поступившей информации о деятельности сердечно-сосудистой системы человека и выработки рекомендаций.*

## ВВЕДЕНИЕ

Развитие медицинских технологий привело к быстрым изменениям в системе электронного здравоохранения. Инновационная и эффективная модель электронного мониторинга здоровья может стать большим подспорьем для интересующихся своим здоровьем людей.

Эта технология помогает врачам лучше диагностировать и лечить пациентов, не находящихся в физической близости, поскольку иногда очень важно обеспечить лечение или лечить пациентов, которые, к сожалению, находятся далеко от медицинских учреждений с высококвалифицированными специалистами, предоставляющих требуемое лечение. В современную эпоху, передовые медицинские технологии значительно влияют на качество нашей жизни. Это можно наблюдать по изменившемуся уровню долголетия. Медицинские технологии — это обширная область, где инновации играют решающую роль в поддержании здоровья.

Существующая медицинская среда в развивающихся странах позволяет пациентам физически появляться для регулярного медицинского осмотра или госпитализировать пациента, при необходимости, для постоянного наблюдения. Система не является гибкой и требует много времени.

Сегодня, в период высоких технологий, различные устройства позволяет пациентам постоянно контролировать свой повседневный образ жизни. Они, как правило, совмещены с наручными часами, чехлами для телефонов или представлены в виде отдельных самостоятельных устройств. Но данные устройства имеют некоторые недостатки. А также еще нет ни одного подобного устройства в отечественном сегменте.

Данная работа представляет возможного конкурента для зарубежных аналогов, которая способна анализировать здоровье пациента в режиме реального времени. Комплекс мониторинга здоровья человека может отслеживать данные о частоте сердечных сокращений, насыще-

ние крови кислородом и кардиомониторинге. С помощью данного приложения можно будет показать результаты лечащему врачу. Потенциальное использование данной системой полезно для диагностики, чтобы иметь представление о состоянии здоровья на основе постоянного контроля пациента. По данным Американской ассоциации лечение сердца в течение рекомендованных 90 минут могут значительно снизить риск смерти или истощения [1].

## I. АНАЛОГИ

На рынке можно встретить ряд устройств, а также приложений, которые могут предоставлять вполне объемную и точную информацию о работе сердечно-сосудистой системы. Ниже приведен обзор ряда устройств и приложений, предоставляющий схожий функционал [2].

Устройство CardioQVARK [3], выполненное в виде чехла для телефонов фирмы Apple. Устройство Withings Move [4] выглядят как обычные классические часы со стрелочками, но все данные с датчиков, установленные с обратной стороны часов, посылаются на мобильное устройство, которое снабжено специальным приложением. Устройство LifeTip [6] выполненное в виде небольшого треугольника, которое крепиться на нижнем белье или одежде человека в области груди.

Вышеописанные устройства, представляют собой одни из самых популярных портативных аппаратов, регистрирующих работу сердечно-сосудистой системы. Все они предназначены для одной цели — считывание каких-либо показателей сердечно-сосудистой системы, таких как пульс, сатурация, ЭКГ. Многие из них имеют различные дополнительные возможности отображения текущего времени до анализа самих данных. Но если учесть определенный ценовой сегмент, например устройства до 100 долларов, то выяснится, что нет устройства, способного самостоятельно анализировать считанные значения и предоставлять различные сове-

ты пользователю. Оно может иметь возможность отправления данных для выполнения анализа профильному специалисту, но не более.

## II. ПРОЕКТ КОМПЛЕКСА

В рамках данной работы ставилась задача — разработать мобильное приложение для одного из подобных гаджетов в виде комплекса мониторинга здоровья сердца человека. Аппаратная платформа комплекса является персональным, мобильным и уникальным устройством, подобного которому не выявлено в отечественном сегменте. Данная разработка представляет возможность анализировать здоровье человека в режиме реального времени, путем отслеживания данных:

- частота сердечных сокращений;
- насыщение крови кислородом;
- кардиомониторинг.

Также комплекс должен обладать следующими параметрами:

- приложение должно быть достаточно просто в обращении;
- данные должны обрабатываться и отображаться на экране мобильного устройства;
- должен производиться первичный анализ полученных данных.

Требования к системному программному обеспечению:

- мобильное устройство должно поддерживать протокол Bluetooth версии 2.0;
- мобильное устройство должно поддерживать операционную систему Android версии 8.0 и выше.

Задача проекта: разработать приложение для получения и визуализации данных о работе сердечно-сосудистой системы с внешнего устройства, а также анализе полученных данных.

Средства для решения задачи: Java, Android, Git, Android Studio, JDK, Android SDK.

С помощью данного устройства, результаты которого, можно будет показать вашему лечащему врачу. Потенциальное использование данной системой полезно для регулярного обследования, анализе и отчетности, чтобы иметь представление о своём состоянии здоровья на основе постоянного контроля. В нашу задачу также входит создание гибкой среды между врачом и пациентом, проверяя их постоянное самочувствие.

В архитектуре проекта предполагается его изменение и масштабирование, т.е. проект разбит на функциональные блоки (модули), благодаря которым можно добавлять новый функционал в проект без изменения или удаления старого. Из-за того, что проект состоит как из аппаратной, так и программной части, то функциональные блоки тоже будут разбиты на программные и аппаратные. В программной части проекта можно выделить следующие блоки:

- блок мобильного приложения; – блок пользовательского интерфейса;

- блок построения графика;
- блок установки соединения;
- блок приема-передачи данных;
- блок обработки данных;
- блок анализа данных;
- блок реляционной базы данных;
- блок модели базы данных;
- блок настройки приложения;
- блок прав доступа приложения.

В аппаратной части проекта можно выделить следующие блоки:

- микроконтроллер;
- блок приема-передачи для микроконтроллера;
- датчик-электрокардиограф;
- датчик частоты сердечных сокращений;
- датчик насыщения крови кислородом (оксигеметр).

Все модули выполняют установленную перед ними задачу. Модули связаны между собой определенными протоколами, с помощью которых они обмениваются данными, образуя в результате всех процессов единое корректно работающее приложение.

## III. ЗАКЛЮЧЕНИЕ

В изложенном выше подходе можно разработать приложение на операционной системе Android позволяющее посредством соединения Bluetooth получать и анализировать данные о сердечно-сосудистой системе человека с аппаратного устройства.

Стоит отметить, что проект может быть продолжен и функционально дополнен при усовершенствовании устройства мониторинга сердечно-сосудистой системы.

## IV. СПИСОК ЛИТЕРАТУРЫ

1. FACTS When Minutes Matter Systems of Care for Acute Cardiovascular Conditions [Электронный ресурс] / Электронные данные. – Режим доступа: <https://www.heart.org/-/media/files/about-us/policyresearch/fact-sheets/acute-care/fact-sheet-systems-of-care-for-acute-cardiovascular-conditions.pdf?la=en> – Дата доступа: 27.09.2021.
2. Портативные ЭКГ аппараты [Электронный ресурс] / Электронные данные. – Режим доступа: <https://www.livemd.ru/tags/portativnye-ekgapparaty/> – Дата доступа: 27.09.2021.
3. Магазин CardioQVARK [Электронный ресурс] / Электронные данные. – Режим доступа: <https://cardioqvark.ru/> – Дата доступа: 27.09.2021.
4. Магазин Withings Move [Электронный ресурс] / Электронные данные. – Режим доступа: <https://www.withings.com/us/en/withings-move> – Дата доступа: 27.09.2021.
5. Магазин ECG Dongle [Электронный ресурс] / Электронные данные. – Режим доступа: <https://www.madgeeks.by/kardiofleshka-ecg-dongle> – Дата доступа: 27.09.2021.
6. Магазин LifeTip [Электронный ресурс] / Электронные данные. – Режим доступа: <https://medgadgets.ru/shop/lifetip-nosimyjbeskontaktnyj-jekg-monitor.html> – Дата доступа: 27.09.2021.

# ПРИМЕНЕНИЕ ГЛУБОКОГО АВТОЭНКОДЕРА ДЛЯ СХЕМЫ СЖАТИЯ ИЗОБРАЖЕНИЯ НА ОСНОВЕ ПАРАУНИТАРНОГО БАНКА ФИЛЬТРОВ

Леченко А. В., Петровский Н. А.

Кафедра информатики,

Кафедра электронных вычислительных средств,

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {lechenko, nick.petrovsky}@bsuir.by

*В работе представлен комбинированный схема сжатия изображений на основе параунитарного банка фильтров в алгебре кватернионов ( $Q$ -ПУБФ) и глубокого автоэнкодера. Рассматривается сжатие изображений оттенков серого с разрешением  $1280 \times 768$ . Предложенный метод состоит из пространственной фильтрации изображения с помощью  $Q$ -ПУБФ с последующим сжатием низкочастотного канала автоэнкодером и квантованием остальных каналов. Результаты показывают, что восстановленные при помощи автоэнкодера изображения лучше по объективным характеристикам ( $PSNR$  и  $SIMM$ ) в сравнении с равномерным квантованием коэффициентного банка фильтров для случая низкой битовой скорости.*

## ВВЕДЕНИЕ

Алгоритмы сжатия изображений играют важную роль в передаче и хранении мультимедиа данных в условиях ограниченной ширины канала или лимитированном объеме хранилища данных. Современные стандарты сжатия изображений, как правило используют банки фильтров для разделения на субполосы, что позволяет более эффективно декоррелировать входные данные [1]. В настоящее время идет поиск методов сжатия изображений с помощью нейронных сетей [2], в том числе глубоких автоэнкодеров [3-4]. Применение нейронных сетей для сжатия изображений обсуждалось и ранее [5], однако комбинирование нейросетевых методов с классическими пространственно-частотными преобразованиями представляют интерес из-за высокой производительности.

В данной работе предлагается схема сжатия изображений, в которой изображение сначала разбивается на частотно-пространственные субполосы с помощью параунитарного банка фильтров ( $Q$ -ПУБФ) на основе алгебры кватернионов [6], после чего низкочастотный канал сжимается с помощью глубокого автоэнкодера [3-4], а остальные каналы квантуются с помощью равномерного квантователя в соответствии с их энергией и выделенным бюджетом бит [7].

## I. СХЕМА СЖАТИЯ

Пусть на вход кодера подается изображение размером  $M \times N$ . С помощью 1D 4-х канального  $Q$ -ПУБФ реализован 2D 16-канальный банк фильтров для декорреляции изображения. Банк фильтров концентрирует перцептуально значимую информацию в низкочастотном (НЧ)-канале для большинства изображений. Далее НЧ-канал делится на 4 равных части, каждая из которых кодируется соответствующим пред-

бученным автоэнкодером. Выходом автоэнкодера является 32-х слойный тензор, который стохастически бинаризуется для получения дискретных значений. Кодовой слой автоэнкодера кодирует содержимое НЧ-канала со скоростью 2 бита на значение исходного канала. Остальные высокочастотные каналы дискретизируются в соответствии с алгоритмом [7], учитывая, что 2 бита на значение уже распределены для НЧ части.

После того как изображение было квантовано подобным образом, НЧ-канал можно декодировать с помощью декодера глубокого автоэнкодера, остальные каналы можно домножить на соответствующий шаг квантования и подать получившиеся изображения на вход банка фильтров синтеза  $Q$ -ПУБФ, на выходе которого будет реконструированное изображение.

## II. СТРУКТУРА АВТОЭНКОДЕРА

Рассмотрим подробнее архитектуру глубокого автоэнкодера. Выбранный автоэнкодер основан на работах [3-4] и состоит из блока энкодера с модулем стохастической бинаризации и блока декодера. Подробная схема автоэнкодера приведена на рисунке 1.

Первые два слоя автоэнкодера увеличивают количество слоев до 128 и ширину и высоту тензора в 4 раза. Далее следует линейный сверточный блок из двух слоев  $\text{LeakyReLU}$  между ними. К выходному тензору этого блока прибавляется значение на входе. Такой сверточный блок применяется три раза к остаточным значениям. Завершается энкодер еще одним сверточным слоем, который уменьшает количество каналов до 32. Полученный выходной тензор передается на вход функции  $\tanh(\cdot)$ , которая отображает значения тензора на интервал от  $[-1, 1]$ , пригодный для стохастической бинаризации. Стохастическая бинаризация вычисляется по формуле:  $b = \text{sign}(x + \psi)$ , где  $\psi$  – слу-

чайная величина с равномерным распределением в диапазоне  $[-1, 1)$ . Бинаризованный тензор и есть кодовый слой. Таким образом, часть изображения  $m \times n$  преобразуется в тензор размеров  $32 \times (n/4) \times (m/4)$  бит, или 2 бита на значение исходного части.

Декодер в основном повторяет структуру энкодера в обратном порядке: два сверточных слоя, которые увеличивают тензор до 128 каналов, три линейных блока свертки, и далее слой увеличивающие ширину и высоту до исходных.

### III. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Для тестирования предложенной схемы, из Youtube-8m [8] были случайно выбраны 1700 изображений оттенков серого в разрешении  $1280 \times 768$ . Изображения были разделены на субполосы каскадом анализа  $Q$ -ПУБФ. Затем часть изображений делилась на обучающую и тестовую выборки. НЧ-каналы обучающей выборки использовались для обучения глубокого автоэнкодера. После того как автоэнкодер был обучен, генерировались восстановленные НЧ-каналы для изображений из тестовой выборки.

Для сравнения метода, в качестве опорного использовался метод [7], поэтому остальные каналы кодировались в соответствии с выделенным бюджетом бит с учетом того, что НЧ-канал потреблял только два бита на значение, т. е., нераспределенные биты на НЧ канал могли распределяться между остальными каналами. Далее полученные квантованные каналы и НЧ-канал, полученный при помощи автоэнкодера, использовались для восстановления изображения каскадом синтеза  $Q$ -ПУБФ, для полученного соотношения считались пиковое отношение сигнал-шум (PSNR) и индекс структурного сходства (SSIM). Сравнение средних значений этих метрик на всех изображениях с аналогичными для метода [7] приведены в таблице 1.

Как видно из таблицы, с ростом количества бит на пиксель (BPP), выделяемых на квантование методом [7], предложенное решение начинает проигрывать. В случае 0.375 BPP, метод [7] выделяет 6 бит на значение для НЧ-канала, что в три раза больше, чем доступно автоэнкодеру.

Перераспределение 4 бит на другие следующие по энергии каналы не позволяет получить заданную точность. Однако предложенная схема дает лучший результат для высоких степеней сжатия. Метрики качества предложенного решения почти в полтора раза выше, когда оба метода используют 0.125 BPP для кодирования изображения и все еще больше, даже если увеличить бюджет бит для метода [7] в полтора раза.

Таблица 1 – Сравнение эффективности сжатия предложенной схемой и методом [7]

| BPP    | SSIM  | SSIM [7] | PSNR | PSNR [7] |
|--------|-------|----------|------|----------|
| 0.125  | 0.834 | 0.444    | 32.1 | 22.7     |
| 0.1875 |       | 0.778    |      | 30.5     |
| 0.375  | 0.839 | 0.938    | 32.3 | 37.1     |
| 0.8    | 0.866 | 0.962    | 33.2 | 40.3     |
| 1.6    | 0.877 | 0.980    | 33.6 | 44.3     |

Из анализа полученных результатов следует вывод, что применение автоэнкодера показывает улучшение объективных характеристик для восстановленных изображений при низких битрейтах в представленном эксперименте.

1. Saramaki T., Bregovic R. Multirate systems and filterbanks //Multirate systems: design and applications. – IGI Global, 2002. – С. 27-85.
2. Ma S. et al. Image and video compression with neural networks: A review //IEEE Transactions on Circuits and Systems for Video Technology. – 2019. – Т. 30. – №. 6. – С. 1683-1698.
3. Theis L. et al. Lossy image compression with compressive autoencoders //arXiv preprint arXiv:1703.00395. – 2017.
4. Toderici G. et al. Variable rate image compression with recurrent neural networks //arXiv preprint arXiv:1511.06085. – 2015.
5. Dony R. D., Haykin S. Neural network approaches to image compression //Proceedings of the IEEE. – 1995. – Т. 83. – №. 2. – С. 288-303.
6. Парфенюк М., Петровский А. А. Параунитарные банки фильтров на основе алгебры кватернионов: теория и применение //Цифровая обработка сигналов. – 2008. – №. 1. – С. 22-36.
7. Петровский Н. А. Оптимальное распределение бит в параунитарном субполосном кодере изображений на основе алгебры кватернионов //Доклады БГУИР. – 2014. – №. 1. – С. 72-77.
8. Abu-El-Haija S. et al. Youtube-8m: A large-scale video classification benchmark //arXiv preprint arXiv:1609.08675 – 2016.

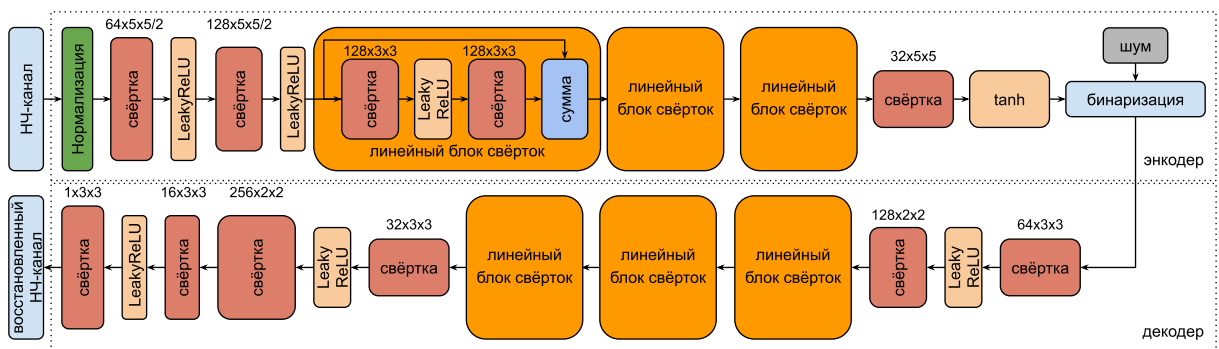


Рис. 1 – Схема глубокого автоэнкодера. LeakyReLU взят с коэффициентом утечки 0.01.

# TREE SHAKING КАК МЕТОД ОПТИМИЗАЦИИ СБОРКИ WEB-ПРИЛОЖЕНИЯ

Мазура А. А., Гуринович А. Б.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь

E-mail: mazura.anastasiya@gmail.com, gurinovich@bsuir.by

*В процессе разработки программного обеспечения использование сборщиков приложений, имеющих по умолчанию установленный tree shaking, не дает заметного результата для оптимизации. Поэтому исследования принципов для эффективной оптимизации web-приложения данным методом актуальны.*

## ВВЕДЕНИЕ

Tree shaking — это термин, обычно используемый в контексте JavaScript для устранения «мертвого» кода. Он опирается на статическую структуру из ES2015 модуля синтаксиса, то есть `import` и `export`. Название и концепция были популяризированы благодаря накопительному пакету модулей ES2015.

Как работает tree shaking:

- Объявляются импорты и экспорты в каждом модуле.
- Сборщик (Webpack, Rollup или другой) во время сборки анализирует дерево зависимостей.
- Неиспользуемый код исключается из итоговой сборки.

К сожалению, для правильной работы tree shaking одной настройки сборщика недостаточно. Чтобы достичь лучшего результата, необходимо учесть множество деталей, а также удостовериться, что модули не были пропущены при оптимизации.

## I. ИСПОЛЬЗОВАНИЕ СИНТАКСИСА ES6 ДЛЯ ИМПОРТОВ И ЭКСПОРТОВ

Использование ES6 импортов и экспортов — первый и важнейший шаг к работающему tree shaking.

Большинство других реализаций паттерна «модуль», включая `common.js` и `require.js`, являются недетерминированными в процессе сборки. Эта особенность не позволяет сборщикам типа Webpack точно определить, что импортируется, что экспортируется и, как следствие, какой код может быть безболезненно удалён.

```
module.exports.foo = function foo() {}
module.exports[someVariable] = function foo() {}
function exportFoo() {
 module.exports.foo = function foo() {}
}
function importFoo() {
 const { foo } = require(baseDirectory + '/foo.js');
}
```

Рис. 1 – Варианты, возможные при использовании `common.js`

При использовании ES6 модулей возможности импорта и экспорта более ограничены:

- импорт и экспорт только на уровне модуля, а не внутри функции;
- имя модуля может быть только статичной строкой, не переменной;
- всё, что импортируется, обязательно должно быть где-то экспортировано.

```
export function foo() {};
import { foo } from './foo';
```

Рис. 2 – Возможности ES6 модулей

Упрощённые правила позволяют сборщикам точно понимать, что было импортировано и экспортировано, и, как следствие, определять, какой код не используется вовсе.

## II. АТОМАРНЫЕ ЭКСПОРТЫ

Webpack, как правило, оставляет экспорты нетронутыми в следующих случаях:

- экспортируется объект с большим количеством свойств и методов;
- экспортируется класс с большим количеством методов;
- экспорт объекта по умолчанию используется для экспорта множества разных функций (Рисунок 3).

```
export default {
 add(a, b) {
 return a + b;
 },
 subtract(a, b) {
 return a - b;
 }
}
```

Рис. 3 – Обе функции будут включены в сборку, даже если используется только одна



Такие экспорты будут либо полностью включаться в сборку, либо полностью удаляться. Значит, в итоге может получиться сборка, содержащая код, который никогда не будет использоваться.

Необходимо сохранять экспорты настолько маленькими и простыми, насколько это возможно. Данный подход позволяет сборщику выкидывать больше кода благодаря тому, что в процессе сборки можно отследить, какая из функций была импортирована и использована, а какая не была.

Практика атомарных экспортов также помогает писать код в более функциональном и направленном на переиспользование стиле, а также избегать использования классов там, где это не оправдано.

### III. ПОБОЧНЫЕ ЭФФЕКТЫ НА УРОВНЕ МОДУЛЯ

При написании модулей многие разработчики упускают важный фактор — влияние побочных эффектов.

```
export function add(a, b) {
 return a + b;
}
export const memoizedAdd = window.memoize(add);
```

Рис. 4 – Webpack не понимает, что делает `window.memoize`, и поэтому не может удалить эту функцию

В примере выше (Рисунок 4) `window.memoize` будет вызвана в момент импорта модуля.

Как это видит Webpack:

- здесь создаётся и экспортируется чистая функция `add` — может быть можно удалить её, если она не будет использоваться позже;
- далее вызывается `window.memoize`, в которую передаётся `add`;
- сборщик не знает, что делает `memoize`, но он точно знает, что `memoize`, возможно, вызовет `add` и создаст побочный эффект;
- поэтому для сохранности сборщик оставит функцию `add` в сборке, даже если её больше никто не использует.

В реальности мы уверены, что `window.memoize` — чистая функция, которая не создаёт никаких побочных эффектов и вызывает `add`, если кто-то использует `memoizedAdd`.

Но Webpack этого не знает и для спокойствия добавляет функцию `add` в итоговую сборку.

```
import { memoize } from './util';
export function add(a, b) {
 return a + b;
}
export const memoizedAdd = memoize(add);
```

Рис. 5 – Исправленная версия кода

Теперь сборщику хватит информации для анализа (Рисунок 5):

- здесь вызывается `memoize` на уровне модуля, это может повлечь проблемы;
- но функция `memoize` пришла из ES6 импорта, нужно взглянуть на функцию в `util.js`;
- действительно, `memoize` выглядит как чистая функция, здесь нет побочных эффектов;
- если никто не использует функцию `add`, можно безопасно исключить её из итоговой сборки.

Важно помнить, что когда сборщик не получает достаточно информации для принятия решения, он пойдёт по безопасному пути и оставит функцию.

Без запуска довольно трудно определить, как сборщик будет оптимизировать конкретный модуль.

Поэтому очень удобно для проверки запускать билд и исследовать итоговую сборку. Обязательно необходимо просмотреть JavaScript-код и убедиться, что в нём не осталось ничего лишнего, что должно было быть удалено с помощью `tree shaking`.

### IV. ЗАКЛЮЧЕНИЕ

Использование исследованных подходов и алгоритмов в процессе разработки веб-приложения позволит эффективно применять `tree shaking`, являющийся неотъемлемой частью сборщика. В свою очередь `tree shaking` эффективно оптимизирует код, уменьшая размер сборки.

### V. СПИСОК ЛИТЕРАТУРЫ

1. Frank Zammetti. Modern Full-Stack Development. Using TypeScript, React, Node.js, Webpack, and Docker. – 2020. – с. 396.
2. Juho Vepsäläinen. SurviveJS - Webpack and React. – 2016. – с. 40-43.
3. Webpack bundle analyzer. / Mode of access: <https://www.npmjs.com/package/webpack-bundle-analyzer> – Date of access: 15.10.2021.
4. DLL Plugin. / Mode of access: <https://github.com/webpack/docs/wiki/list-of-plugins#dllplugin> – Date of access: 15.10.2021.
5. Thee shaking / Mode of access: <https://webpack.js.org/guides/tree-shaking/> – Date of access: 16.10.2021.

# ИЗВЛЕЧЕНИЕ КЛЮЧЕВЫХ СЛОВ: ГРАФООРИЕНТИРОВАННЫЙ ПОДХОД

Мазура И. А., Гуринович А. Б.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: irynamazura22@gmail.com, gurinovich@bsuir.by

*Задача автоматического выделения ключевых слов из текста возникает во многих областях. Актуально, как исследование существующих алгоритмов, так и анализ методов, позволяющих их оптимизировать.*

## ВВЕДЕНИЕ

Извлечение ключевых слов является важной задачей, лежащей на стыке таких областей знаний, как интеллектуальный анализ текста (Text Mining), информационный поиск (Information Retrieval) и обработка естественного языка (Natural Language Processing). Под ключевыми словами понимаются слова или фразы, которые наилучшим образом характеризует содержание исследуемого текста. Задача автоматического определения ключевых слов представляет собой необходимый этап обработки текста для решения таких задач, как: создание и развитие терминологических ресурсов, автоматический информационный поиск, аннотирование, классификация и кластеризация документов, суммаризация текста и другие.

### I. КРАТКИЙ ОБЗОР ПОДХОДОВ К ИЗВЛЕЧЕНИЮ КЛЮЧЕВЫХ СЛОВ

Анализ литературы в области извлечения ключевых слов выявил большое число методов и их модификаций, однако общепринятой в научном сообществе классификации данных подходов на текущий момент не существует. Действительно, решение задачи автоматического выделения ключевых слов ведется одновременно по двум направлениям. С одной стороны, методы различаются по типу математического аппарата распознавания ключевых слов (статистические, методы на основе машинного обучения, структурные), с другой — по типу используемых (или не используемых) лингвистических ресурсов (словари, онтологии и тезаурусы, корпуса текстов). Наиболее простым статистическим методом извлечения ключевых слов является метод ранжирования всех словоформ по частоте. При подсчете частоты употребления ключевого слова учитываются все его словоформы в тексте. Создаваемые на основе данного подхода алгоритмы являются недостаточно точными, т.к. признак частотности ключевых слов не является преобладающим [1].

Для повышения корректности автоматического извлечения ключевых слов, статистический метод дополняется лингвистическими про-

цедурами (морфологическим, синтаксическим или семантическим анализом). Такие методы могут требовать или не требовать корпусов текстов. Использование корпуса текстов получило достаточно широкое распространение, однако отсутствие таких корпусов для каждой конкретной предметной области в реальной жизни делает применение таких корпусных моделей и методов весьма проблематичным. Методы на основе машинного обучения рассматривают задачу извлечения ключевых слов как задачу классификации — вычисление вероятности отнесения слова к ключевому на основе обучающей выборки — корпуса документов с размеченными ключевыми словами. В основе структурных методов лежит представление о тексте как системе семантически и грамматически взаимосвязанных элементов-слов, которые, в свою очередь, характеризуются набором лингвистических признаков. Здесь могут быть выделены два подкласса — графовые и синтаксические (шаблонные) методы.

### II. ГРАФООРИЕНТИРОВАННЫЙ ПОДХОД

Выбор того или иного алгоритма извлечения ключевых слов обуславливается в первую очередь естественным языком, спецификой исследуемой темы, объемом анализируемого текста. Графовые модели представляют большой интерес для области обработки естественного языка благодаря своей универсальности (не зависят от естественного языка) и эффективности основанных на них алгоритмов. Графовые методы не предполагают использование лингвистических ресурсов для настройки критериев принятия решений при распознавании ключевых слов. Вместо этого, в работе алгоритмов подразумевается контекстно-независимое выделение ключевых слов, что является оптимальным решением для гомогенных по функциональному стилю корпусов текстов, например, научных работ или нормативных актов, а также работ, посвященных новым, развивающимся темам, для которых не существует разработанных словарей, тезаурусов и т.п.

Итак, в основе графовых моделей лежит процедура построения графа, в вершинах кото-

рого стоят лексические единицы (слова или предложения), а отношения между ними представлены в виде ребер графа. Отношение между лексическими единицами может быть основано на различных принципах, наиболее распространенными из которых являются:

- Отношение совместной встречаемости — связанные слова встречаются в тексте внутри окна фиксированного размера; связаны все слова внутри предложения, параграфа или документа.
- Семантическое отношение — связанные слова имеют одинаковое значение, синонимы, антонимы, омонимы и др.

Для вершин полученного графа вычисляется мера центральности как индикатор определения наиболее значимых вершин внутри графа. Центральность вершины  $v$  — это мера, отражающая то, насколько вершина  $v$  участвует в процессе распространения информации между остальными вершинами в графе. В области извлечения ключевых слов используются различные меры центральности, на основе которых производится ранжирование слов текста [3].

### III. АЛГОРИТМ TEXTRANK

В основе TextRank лежит процедура построения взвешенного графа, в вершинах которого стоят лексические единицы (слова или предложения), а ребра взвешены в соответствии с силой связи между ними. После того, как произведена предобработка текста, производится построение взвешенного неориентированного графа  $G = (V, E)$ , где  $V$  — множество уникальных лексических единиц (вершины);  $E$  — множество связей между ними (ребра) [2].

В качестве меры связи между словами TextRank использует отношение совместной встречаемости: две вершины соединяются ребром, если их лексические единицы встречаются вместе внутри окна из  $N$  слов,  $N \in [2, 10]$ . Формула для определения веса каждого ребра:

$$w_{ij} = \begin{cases} 1 - \frac{d(w_i, w_j) - 1}{(N - 1)}, & \text{если } d(w_i, w_j) \in (0, N) \\ 0, & \text{если } d(w_i, w_j) \geq (0, N) \end{cases}$$

На следующем этапе по итеративной формуле вычисляется TextRank (TR), получаемый случайным блужданием для каждой вершины из  $V$ :

$$TR(V_i) = (1 - d) + d \cdot \sum_{v_k \in \text{In}(V_i)} \frac{w_{ij}}{\sum_{v_k \in \text{Out}(V_i)} w_{jk}} TR(V_j),$$

Начальное значение TR для каждой из вершин предполагается равным 1. Алгоритм повторяется до достижения порогового уровня сходимости.

В соответствии с итоговыми значениями TR вершины графа ранжируются, отбираются  $T$  «лучших» вершин (с наибольшим значением TR). Ключевые фразы получают путем извлечения из текста последовательностей, состоящих из  $T$ -лучших слов [4].

### IV. ЗАКЛЮЧЕНИЕ

Алгоритм TextRank показывает адекватные результаты на обработке текстов сравнительно большого размера. В работе алгоритма были выявлены определенные сложности в сборке словосочетаний: «склеивание» слов в фразы происходит в полуручном режиме (необходимо участие эксперта), так как автоматическая сборка, во-первых, способна вывести только наборы слов-лемм, во-вторых, недостаточность ограничений на этапе сборки словосочетаний приводит к появлению заметного количества бессмысленных строк, собранных из вершин с большим весом. Основное решение проблемы состоит в учете вложенности и частоты встречаемости терминов в тексте.

Несмотря на большое число методов извлечения ключевых слов, до настоящего времени не разработана последовательная методика обнаружения ключевых слов человеком. Экспериментально подтверждено, что эта операция выполняется людьми интуитивно и является личностно обусловленной. Так, в случае извлечения ключевых слов из небольшого числа документов, результаты будут чувствительны к количеству слов, которое — было принято решение — отсечь от общего числа слов-кандидатов в ключевые слова. Употребляя одни и те же слова в разных словосочетаниях и контекстах, авторы текстов уходят от четких границ терминов. Тем самым размывается их содержание, исчезает терминологический статус.

Актуальным на данный момент является вопрос о составлении объективной методики извлечения ключевых слов.

### V. СПИСОК ЛИТЕРАТУРЫ

1. Beliga, S., Martincic-Ipsic, S., and Meštrović, A. An Overview of Graph-Based Keyword Extraction Methods and Approaches. *Journal of Information and Organizational Sciences*, 2015,39(1).
2. Page, S., Brin, S., Motwani, R., and Winograd, T. The PageRank Citation Ranking: Bringing Order to the Web. Technical Report. Stanford: Stanford University, 1998.
3. Automatic Keyword Extraction from Individual Documents [Электронный ресурс]. – Режим доступа: [https://www.researchgate.net/publication/227988510\\_Automatic\\_Keyword\\_Extraction\\_from\\_Individual\\_Documents](https://www.researchgate.net/publication/227988510_Automatic_Keyword_Extraction_from_Individual_Documents).
4. Understand TextRank for Keyword Extraction by Python [Электронный ресурс]. – Режим доступа: <https://towardsdatascience.com/textrank-for-keyword-extraction-by-python-c0bae21bce0>.

# АВТОМАТИЗИРОВАННЫЕ ДИАЛОГОВЫЕ СИСТЕМЫ

Навроцкий А. А., Кривальцевич Е. В.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: navrotsky@bsuir.by, elena.krivaltsevich@gmail.com

*Обработка естественных языков, а, в частности, создание автоматизированных диалоговых систем, является актуальной задачей в различных сферах деятельности, начиная от сферы развлечений, заканчивая документооборотом. Чаще всего используются Hidden Markov Model для создания диалоговых систем, предназначенных для одного пользователя. Тем не менее, некоторые исследователи применяют другие методы, в том числе, для создания полилоговых систем.*

## ВВЕДЕНИЕ

Разработка и внедрение автоматизированных диалоговых систем является отражением постепенной, но неуклонной цифровизации и автоматизации взаимодействия пользователей с такими областями как предоставление специализированных услуг и консультаций, помощь в выборе товара, регистрация и сопровождение на сайте, предоставление психологической поддержки, обучение, досуг и многими другими.

Создание систем по принципу вопрос-ответ (Question answering) является комплексной и сложной задачей. При разработке чат-ботов часто используются Natural Language Processing методы и алгоритмы. Предварительно необходимо определить тему разговора (topic entity). Далее – корректно определить последовательность именованных сущностей в предложении, маркировать последовательности (sequence labelling), правильно выделить части речи, осуществить парсинг, выделить именованные сущности.

Чат-боты, разработанные и используемые в мире на данный момент, реализованы при помощи широкого спектра моделей машинного обучения и нейронных сетей. Наиболее часто используемыми моделями являются Hidden Markov Model (HMM), Maximum Entropy Markov Model (MEMM), Conditional Random Fields (CRF), structured SVM. Также чат-боты могут дополнительно регулироваться списками правил для более точного выполнения поставленных задач.

Чат-боты могут быть реализованы как отдельные мобильные приложения и программное обеспечение, так и быть встроенными в сайт, социальную сеть или мобильное приложение.

## I. ПРИМЕРЫ РЕАЛИЗАЦИИ

Как показано в статье исследователей из Carnegie Mellon University, разработкой чат-ботов, структура которых позволяет сразу многим пользователям участвовать в разговоре с ними (по сути – создавать полилог), занимаются только отдельные немногочисленные исследователи. В данной работе авторы создали чат-бот, который «растёт» и обучается на речевых моде-

лях игрового сообщества в Twitch. Данный чат-бот имеет три стадии «взросления». В результате своей работы исследователи узнали, что пользователи, не смотря на то, что были подвержены к обучению чат-бота некорректным, и, порой, обидным выражениям, всё же – были весьма заинтересованы в его обучении. Сообщество развило некоторый самоконтроль и, в конце концов, стало воспринимать чат-бота более как участника разговора и менее как программное обеспечение, которому требуется обучение. [1]

Исследователи из итальянского университета Università Ca' Foscari Venezia подняли вопрос разработки диалоговой системы узкой направленности, а именно для работы с посетителями аэропорта в Венеции. Данные исследователи использовали готовое решение Microsoft Azure Bot с двумя дополнительными расширениями функциональности – LUIS и QnA Maker. LUIS позволяет выделять именованные сущности, определять ошибки произношения и исправлять их при обработке. QnA Maker используется при обработке большого количества текстовых документов с частыми вопросами, а также ссылок и мануалов по продукту для составления упорядоченной информации в форме «вопрос – ответ». Также исследователи подключили локальные данные аэропорта, такие как информация о времени прибытия и отправления самолетов, информация о местных транспортных службах, информация о парковочных местах и т.д. [2]

Задачей проекта корейских исследователей из Seoul National University была помощь и облегчение инициирования диалога между двумя потенциальными кандидатами при помощи исследования недавних постов их социальных сетей. Структура данного чат-бота затрагивает использование социальной сети Instagram, AWS EC2 server, NLP модуля для корейского языка KoNLPy и Word2Vec. Также авторы использовали список правил для регулирования длительности обсуждения каждой темы. Исследователи резюмировали, что участники эксперимента, которые использовали BlahBlahBot, отмечали значительное субъективное облегчение как в инициации разговора, так и в ходе беседы в целом. Так-

же чат-бот помог избегать нежелательных тем. [3]

Австралийские исследователи из University of New South Wales изучали вопрос эффективности применения Recurrent Neural Networks при создании диалоговых систем в сфере банковских услуг. Поскольку Hidden Markov Models ограничены в своем использовании тем, что более успешно могут быть применены к моноязыковым текстам определенного домена, авторы предлагают использовать attention-based seq2seq модель для Deep Recurrent Neural Networks. Также авторы решают проблему отсутствия данных для создания ответа чат-ботом. В обычном случае поиск ответа производится в заранее созданных шаблонах ответов, при неудаче – в QA секции продукта. Если же нет необходимой информации и там, то алгоритм переходит к поиску информации на сайтах, список которых заранее сформирован. В противном случае, используется DRNN для генерации ответа, опираясь на вопрос пользователя. [4]

Российские исследователи из Финансового университета при Правительстве Российской Федерации обсуждают вопрос влияния ИТ-технологий, в том числе, чат-ботов на эффективность работы в компании. Исследователи опираются на «Тектологию» Богданова А.А., тем самым обосновывая необходимость внедрения чат-ботов в административные процедуры в качестве медиатора между сотрудником и постоянно растущим количеством данных в рамках продукта. [5]

## II. ПРОБЛЕМЫ РАЗРАБОТКИ

Исследователи полилогового системы, «взрослеющей» постепенно, столкнулись с переобучением, а также с обучением на нежелательных выражениях, что требует дополнительного времени на разработку правил, которые могли бы очищать корпус текста, использующийся для обучения данного типа чат-бота. [1]

Для исследователей чат-бота как сервиса для посетителей аэропорта фундаментальными параметрами качества являются качество перевода речи в текст и качество общения с чат-ботом через get и post запросы. Тем не менее, стандартный пакет Microsoft Azure Bot может недостаточно качественно обрабатывать речь пользователя, и потому, выдавать нерелевантную информацию. Поэтому исследователи применяют Mozilla Webkit Speech Recognition для обработки речевых данных. Дополнительную задачу перевода текста в речь при помощи Mozilla Speech Synthesis Utterance им не удалось решить из-за несовершенства произношения в данном инструменте. [2]

Корейские исследователи предпочли использовать Word2Vec вместо BERT, поскольку BERT предоставляет несколько векторных пред-

ставлений для одного и того же слова, что не подходило для их эксперимента, а также потому что вычислительная стоимость BERT выше и это могло отрицательно влиять на скорость выдачи текстового сообщения чат-ботом. Некоторые пользователи остались недовольны тем, что при автоматической смене темы разговора чат-ботом невозможно было выбрать опцию продления разговора по предыдущей теме. [3]

Метод, приведенный австралийскими исследователями, обладает значительными преимуществами в виде отсутствия необходимости в обучении в рамках строго определенных доменов или языков. Тем не менее, предложенная разветвленная структура генерации ответов пользователю может приводить к ошибкам определения сути запроса пользователя ввиду отсутствия определенного шаблона, отсутствия действительно необходимого сайта в списке или базе данных «вопрос-ответ» продукта. Что, в свою очередь, будет вызывать недовольство пользователей и низкую загруженность чат-бота.

## III. ЗАКЛЮЧЕНИЕ

Область создания автоматизированных систем «вопрос-ответ», виртуальных помощников, чат-ботов является одной из самых сложных задач обработки естественно-языковых текстов и речевых сигналов. Изучение материалов исследователей из разных стран показал, что на данный момент часто используются НММ, но наряду с этим, могут быть использованы RNN, подходы, основанные на правилах, или разнообразные комбинации существующих решений с дополнительными разработками в рамках необходимого домена. Чат-боты используются в самых разных областях – существует ряд решений для документооборота и информирования, для туристической сферы, образовательной, развлекательной и других.

1. Seering, J. It Takes a Village: Integrating an Adaptive Chatbot into an Online Gaming Community / J. Seering, M. Luria, C. Ye [et al.] // CHI 2020, April 25–30, Honolulu, HI, USA -- 2020.
2. Carisi, M. Design and implementation of an airport chatbot / M. Carisi, A. Albarelli, F. L. Luccio // GoodTechs '19, September 25–27, Valencia, Spain -- 2019.
3. Shin, D. BlahBlahBot: Facilitating Conversation between Strangers using a Chatbot with ML-infused Personalized Topic Suggestion / D. Shin, S. Yoon, S. Kim [et al.] // CHI '21 Extended Abstracts, May 8–13 / Japan. – Yokohama, 2021.
4. Nuruzzaman, M. Identifying Facts for Chatbot's Question Answering via Sequence labelling Using Recurrent Neural Networks / M. Nuruzzaman, O. Kh. Hussain // ACM TURC 2019, May 17–19 / China – Chengdu 2019.
5. Романова, Е. В. ЧАТ-БОТЫ КАК ЭЛЕМЕНТ УПРАВЛЕНИЯ СИСТЕМОЙ / Е. В. Романова, А. В. Двигубский // «Хроноэкономика». — 2019. -- №7. -- Р. 94–99.

# АЛГОРИТМЫ И МЕТОДЫ УПРАВЛЕНИЯ ПЛАТЕЖНЫМ АГРЕГАТОРОМ

Оберемко М. И.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: oberemko.maxim@gmail.com

*В данной работе рассматривается применение протокола 3-D Secure, в качестве эффективного метода управления безопасностью платежного агрегатора при проведении онлайн операций с кредитными и дебетовыми картами.*

## ВВЕДЕНИЕ

Основополагающим принципом в работе платежного агрегатора является безопасность и целостность персональных данных при проведении онлайн-платежей. В качестве ключевого и незаменимого метода управления безопасностью следует применять протокол 3D Secure (Three Domain Secure). Протокол основан на принципе верификации подлинности через 3 домена. Первый домен – домен банка-эмитента, который выпустил карту, используемую в операции. Второй домен – домен банка-эквайера, который принимает денежные средства. Третий домен – домен совместимости, который представляет собой инфраструктуру, предоставляемую платежной системой при проведении онлайн-платежа.

Кроме обеспечения дополнительного уровня защиты от мошенничества, в случае использования 3D Secure происходит «Перенос ответственности» за мошеннический платеж, при котором вся ответственность переходит от продавца к банку-эмитенту, выпустившему карту. Данное преимущество использования протокола позволяет бизнесу провизитировать конверсию.

## I. АЛГОРИТМ УПРАВЛЕНИЯ ПРОТОКОЛОМ 3D SECURE 1.0

В настоящее время подавляющее большинство банков и платежных систем используют версию 1.0.2 при проведении онлайн CNP-платежей (Card Not Present), запрашивающих OTP-код (One Time Password). Данный протокол разработан на основе XML.

При инициализации транзакции в системе платежного агрегатора запускается алгоритм. В первую очередь осуществляется CRReq-запрос (Card Range Request). Данный запрос необходим, чтобы найти банк-эмитент проверяемой карты и получить CRR из домена взаимодействия. Следующим шагом агрегатор отправляет VeReq-запрос (Verification Request), который содержит информацию о торговце и номер карты плательщика. Этот запрос отправляется банку-эмитенту для проверки того, что 3DS для данной карты включен и ее можно использовать для опла-

ты. После получения ответа VeRes (Verification Response), в котором наиболее важным параметром является URL-адрес, который указывает, где находится сервер эквайера и куда необходимо отправить PaReq. PaReq (Payment Request) – запрос на оплату, в котором передаются данные продавца, информация о платеже и URL-адрес платежного агрегатора, на который будет возвращен плательщик в конце процесса аутентификации 3D Secure. Запрос на оплату выполняется посредством перенаправления на сервер эквайера через браузер плательщика. На стороне эквайера плательщик вводит одноразовый код и возвращается на сайт платежного агрегатора вместе с результатом проверки PaRes (Payment Response). После получения успешного статуса верификации платежный агрегатор совершает запрос в банк эквайер на списание денежных средств.[3]

Несмотря на то, что данную версию протокола используют больше всего, она имеет ряд недостатков и проблем:

1. Протокол поддерживает только взаимодействие через браузерный интерфейс.
2. Верификация держателя карты осуществляется только с помощью смс-кодов.
3. Из-за использования формата XML данная версия уязвима к атакам типа XXE (XML external entity).
4. Потенциальная атака на магазин торговца из-за выполнения PaReq в формате перенаправления.

## II. ПРОТОКОЛ 3D SECURE 2.0

Из-за недостатков протокола 3D Secure 1.0 была создана усовершенствованная версия протокола - 3D Secure 2.0, который развивает EMVCo – организация, созданная международными платежными системами с целью разработки международных стандартов для чиповых карт и операций с ними.

В обновленном протоколе добавили гибкую поддержку различных устройств и каналов. Обеспечили более плавное и последовательное взаимодействие с плательщиком по нескольким каналам оплаты, включая оплату в браузере мобильного телефона, платежи в приложении

ях и платежи через цифровой кошелек. Улучшили пользовательский опыт. Обеспечив продавцам возможность лучше интегрировать процесс аутентификации в процесс покупок, предоставляя держателям карт быструю, простую и удобную аутентификацию при высоком уровне безопасности. В отличие от статических паролей, в 3D Secure 2.0 используются методы динамической аутентификации, такие как биометрия и аутентификация на основе токенов. Улучшили обмен данными для борьбы с мошенничеством и снижения препятствий. В протоколе 2.0 существует 2 варианта аутентификации:

1. Аутентификация с вводом одноразового пароля.
2. Беспрепятственная аутентификация.[1]

Беспрепятственная аутентификация позволяет эмитентам одобрить транзакцию, не требуя ручного ввода данных от владельца карты. Это достигается с помощью так называемой «аутентификации на основе рисков» (RVA). RVA работает, собирая набор данных о держателях карт во время транзакции и передавая их банку-эмитенту и его серверу, который затем сравнивает собранные данные с предыдущими (историческими) данными о транзакциях держателя карты для вывода значения риска мошенничества, соответствующего новой транзакции.

### III. АЛГОРИТМ УПРАВЛЕНИЯ ПРОТОКОЛОМ 3D SECURE 2.0

В обновленной версии протокола 3D сервер платежного агрегатора взаимодействует напрямую в основном с корневым сервером платежной системы (Visa, MasterCard, Maestro). Перед началом работы алгоритма 3D сервер должен запросить у сервера платежной системы информацию о диапазонах номеров карт, которые поддерживают версию 2.0, с помощью подготовительного запроса (PReq). Кроме того 3D должен регулярно обновлять информацию о диапазонах. Данные сообщения не являются частью основного алгоритма.[2]

Алгоритм запускается в отдельном изолированном потоке и проверяет принадлежность карты плательщика к диапазонам, которые сохранены в 3D сервере платежного агрегатора. После подтверждения версии 3D сервер отправляет зашифрованный авторизационный запрос (AReq) на сервер платежной системы. В авторизационном запросе содержатся данные о торговце, покупке и информация о плательщике, например публичные данные его браузера. Именно

на основании этих данных банк эмитент может разрешить беспрепятственную аутентификацию. Если в ответе на авторизационный запрос банк подтвердил принадлежность карты плательщику, то алгоритм завершает верификацию и запускает следующий алгоритм в изолированном потоке, которые выполняет платежный запрос.

В случае, если банку эмитенту не хватило предоставленных данных, то алгоритм продолжает работу и выполняет дополнительный запрос верификации (CReq). Данный запрос, как и при использовании первой версии протокола, выполняется посредством перенаправления плательщика на страницу банка эмитента через его браузер. Как только плательщик пройдет дополнительную проверку корневой сервер платежной системы отправляет результаты на 3D сервере платежного агрегатора. После чего алгоритм завершает верификацию и запускает следующий алгоритм в изолированном потоке, которые выполняет платежный запрос.

### ЗАКЛЮЧЕНИЕ

Рассмотренный метод управления безопасностью платежного агрегатора позволяет установить безопасный канал обмена данными, работающий в режиме реального времени, по которому будет передаваться намного больше данных о транзакции для более точной аутентификации покупателя, увеличится скорость совершения оплаты, поскольку аутентификацию с помощью пароля будут проходить не все транзакции, а только некоторая их часть. Описанный алгоритм управления протоколом 3D Secure 2.0 реализует все нововведения протокола, для обеспечения быстрых и надежных онлайн-платежей через банковские карты.

### СПИСОК ЛИТЕРАТУРЫ

1. EMV3-D Secure Protocol and Core Functions Specification [Electronic resource] / EMVCo LLC. – United States, 2017. – Mode of access: <https://www.emvco.com/emv-technologies/3d-secure/core-functions-specifications>. – Date of access: 17.10.2021.
2. 3-D Secure Browser Flow Best Practices [Electronic resource] / EMVCo LLC. – United States, 2021. – Mode of access: <https://www.emvco.com/emv-technologies/3d-secure/browser-flow-best-practices>. – Date of access: 17.10.2021.
3. Requirement Numbering Scheme and Error Processing [Electronic resource] / EMVCo LLC. – United States, 2021. – Mode of access: <https://www.emvco.com/emv-technologies/3d-secure/requirement-numbering-scheme-and-error-processing>. – Date of access: 17.10.2021.

# НЕПАРАМЕТРИЧЕСКАЯ КЛАССИФИКАЦИЯ МНОГОМЕРНЫХ НАБЛЮДЕНИЙ НА ОСНОВЕ ЯДЕРНЫХ ОЦЕНОК ПЛОТНОСТЕЙ С ПРЯМОУГОЛЬНЫМ ЯДРОМ

Паланевич А. С., Жук Е. Е.

Кафедра математического математического моделирования и анализа данных,

Белорусский государственный университет

Минск, Республика Беларусь

E-mail: apalanevich@gmail.com, zhukee@mail.ru

*В работе рассматривается задача статистической классификации с применением байесовского решающего правила и прямоугольного ядра для оценивания плотностей распределения.*

## ВВЕДЕНИЕ

Статистическая классификация данных является одной из самых известных задач прикладного анализа данных. Она часто возникает в таких областях, как медицина, экономика и т.д. Для решения данной проблемы существует множество методов и алгоритмов, один из которых я бы хотел рассмотреть в своей работе.

Для изучения был выбран метод, основанный на ядерной оценке плотности распределения. В качестве ядра использовалось прямоугольное ядро, так как оно является довольно простым и позволяет особым образом интерпретировать полученные с его помощью результаты.

## I. ОПИСАНИЕ МОДЕЛИ

Пусть регистрируются случайные наблюдения  $x = x(w) \in R^N$  над объектами  $w \in \Omega$ , принадлежащими к  $L$  классам  $\{\Omega_1, \Omega_2, \dots, \Omega_L\}$ :

$$\Omega_i \cap \Omega_j = \emptyset, i \neq j, i, j = 1, 2, \dots, L;$$

$$\bigcup_{i=1}^L \Omega_i = \Omega;$$

Обозначим истинный номер класса, к которому принадлежит объект  $w$ , через  $d^0(w)$ . Этот номер является дискретной случайной величиной со следующим распределением:

$$P(d^0(w) = i) = \pi_i, i = 1, 2, \dots, L;$$

$$\pi_1 + \pi_2 + \dots + \pi_L = 1;$$

Здесь  $\{\pi_i, i = \overline{1, L}\}$  – априорные вероятности классов. В рамках каждого из классов наблюдение  $x(w)$  описывается условной плотностью распределения:

$$p_i(x) = p(x|d^0(x) = i), i = 1, 2, \dots, L, x \in R^N;$$

## II. ПОСТАНОВКА ЗАДАЧИ И ПОСТРОЕНИЕ ОЦЕНОК

Поставим перед собой задачу оценки номера класса для нового наблюдения по имеющейся классифицированной выборке (задача дискриминантного анализа). Для этого будем пользоваться байесовским решающим правилом (БРП) [1]:

$$\hat{d}^0(x) = \arg \max_{i=1,2,\dots,L} (\pi_i p_i(x)), x \in R^N,$$

где  $\hat{d}^0(x)$  – оценка номера неизвестного класса для наблюдения  $x$ . Мы имеем дело с нерандомизированным решающим правилом [1]. Однако для пользования этим правилом необходимо знать  $\{\pi_i, p_i(x), i = 1, 2, \dots, L\}$ . Так как их точные значения неизвестны, то укажем способ построения оценок для этих величин.

Пусть  $X = \{x_1, x_2, \dots, x_n\} \in R^{nN}$  – классифицированная выборка и  $n_k$  – число наблюдений из выборки, которые относятся к классу с номером  $k$ . Тогда построим оценку для априорных вероятностей [1]:

$$\hat{\pi}_k = \frac{n_k}{n}, k = 1, 2, \dots, L. \quad (1)$$

Условные плотности распределения будем оценивать с помощью ядерных оценок с прямоугольным ядром. Пусть  $\Gamma(x)$  –  $N$ -мерный параллелепипед с центром в точке  $x$  и сторонами  $h_i \in R, i = 1, 2, \dots, N$ , с «объемом»  $V = \prod_{i=1}^N h_i$ . Вводя функцию-индикатор  $I_{\Gamma(x)}(y)$ , равную единице, если  $y \in \Gamma(x)$ , и нулю – в противном случае, оценки плотностей запишем следующим образом [2]:

$$\hat{p}_k(y) = \frac{1}{n_k V} \sum_{j=1}^n I_{\Gamma(y)}(x_j) \delta_{k, d^0(x_j)}; \quad (2)$$

$$x_j \in X, j = 1, 2, \dots, n, k = 1, 2, \dots, L, y \in R^N.$$

Тогда для оценки номера класса, к которому принадлежит новое наблюдение  $x^*$ , получаем следующее подстановочное БРП [1]:



$$\hat{d}(x^*) = \arg \max_{k=1,2,\dots,L} (\widehat{\pi}_k \widehat{p}_k(x^*)), x^* \in R^N. \quad (3)$$

### III. ИНТЕРПРЕТАЦИЯ РЕЗУЛЬТАТОВ

Теперь сформулируем и докажем теорему, придающую построенному БРП содержательный смысл.

**Теорема.** Пусть по обучающей выборке  $X = \{x_1, x_2, \dots, x_n\} \in R^{nN}$  построены упомянутые выше оценки параметров модели  $\{\widehat{\pi}_k, \widehat{p}_k(x), k = 1, 2, \dots, L\}$  из (1), (2). Тогда новое наблюдение  $x^* \in R^N$ , классифицируемое с помощью подстановочного БРП (3), относится к классу  $\Omega_d$  с номером  $d$ , когда количество наблюдений из выборки  $X$ , лежащих в  $\Gamma(x^*)$  и принадлежащих этому классу, является наибольшим среди остальных классов.

**Доказательство.** Для простоты рассуждений предположим, что максимум в БРП достигается на одном классе с номером  $d$ . Тогда:

$$\hat{d}(x^*) = d \Leftrightarrow \widehat{\pi}_d \widehat{p}_d(x^*) > \widehat{\pi}_k \widehat{p}_k(x^*), k = \overline{1, L}, k \neq d.$$

Подставляя полученные ранее оценки параметров модели (1), (2) и сокращая множители в дробях, получаем:

$$\sum_{j=1}^n I_{\Gamma(x^*)}(x_j) \delta_{d, d^0(x_j)} > \sum_{j=1}^n I_{\Gamma(x^*)}(x_j) \delta_{k, d^0(x_j)}$$

$$\sum_{j=1}^n I_{\Gamma(x^*)}(x_j) (\delta_{d, d^0(x_j)} - \delta_{k, d^0(x_j)}) > 0.$$

Это неравенство возможно лишь в случае, когда количество наблюдений  $x$ , на которых функция  $\delta_{d, d^0(x)}$  обращается в единицу (то есть  $d = d^0(x)$ ) больше, чем количество аналогичных наблюдений для функции  $\delta_{k, d^0(x)}$ . Это как раз и соответствует сформулированной выше интерпретации построенного БРП.

**Замечание.** В случае, когда максимум подстановочного БРП достигается на нескольких классах (когда в параллелепипеде вокруг нового наблюдения сразу несколько классов имеют

наибольшую частоту), можно пропорционально увеличить стороны  $\Gamma(x^*)$  до тех пор, пока количество элементов какого-либо класса не станет наибольшим. Аналогично можно поступить, когда в  $\Gamma(x)$  вообще не попало ни одно наблюдение.

**Замечание.** Здесь в качестве параметров классификатора выступают стороны параллелепипеда  $h_i, i = 1, 2, \dots, N$ . Их можно выбрать, например, из тех же соображений, что используются при выборе размеров ячейки для построения гистограмм (правило Стерджеса или правило квадратного корня). В таком случае для каждого измерения длина стороны параллелепипеда будет своей.

**Замечание.** Вместо параллелепипеда можно рассматривать  $N$ -мерный куб, то есть положить  $h_i = h, i = 1, 2, \dots, N$ . В таком случае для выбора длины стороны удобно взять сетку  $h^{(1)}, h^{(2)}, \dots, h^{(m)}$  и посмотреть, на каком из значений точность классификатора будет наибольшей.

### ЗАКЛЮЧЕНИЕ

Таким образом, мы получили довольно интересную и интуитивно понятную интерпретацию решающего правила для статистической классификации с использованием прямоугольных ядерных оценок. Это делает построенный метод довольно простым для применения на практике.

Также хочется заметить, что полученный результат, в каком-то смысле, роднит исследуемый в этой работе метод с классификацией наблюдений на основе оценок плотностей с применением гистограмм с прямоугольными ячейками. Основное отличие заключается в том, что в гистограммном случае ячейки имеют фиксированные позиции, а в случае ядерных оценок ячейка строится вокруг нового наблюдения.

### СПИСОК ЛИТЕРАТУРЫ

1. Жук, Е. Е., Харин, Ю. С. Математическая и прикладная статистика // учебное пособие. – Минск, БГУ, 2005. – с. 192–208.
2. Multivariate Kernel Smoothing and its Applications / Jose E. Chacon, Tam Duong – Taylor & Francis Group, LLC, 2018.

# АЛГОРИТМ РЕГУЛИРОВАНИЯ ЦЕНЫ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ

Папкович А. С.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: pasalex94@gmail.com

*Рассматривается алгоритм регулирования цены в условиях неопределенности. Алгоритм состоит из алгоритма принятия решений в условиях неопределенности и нейросети обученной на поиск и анализ необходимой информации.*

**Ключевые слова:** Алгоритм принятия решений, неопределенность, нейросеть, регулирование цены

## ВВЕДЕНИЕ

Для прогнозирования и регулирования цен на продукт предлагается использовать алгоритм регулирования цены в условиях неопределенности, работа которого основана на алгоритме принятия решений в условиях неопределенности, а так же использует нейросеть направленную на анализ новостей и других источников информации. Алгоритм позволит скорректировать цену на прогнозируемый период, на основе графиков предыдущих цен и анализа новостей связанных с этим продуктом.

## I. АКТУАЛЬНОСТЬ

Современная экономика диктует участникам торгово-экономических отношений определенные правила взаимодействий. Одним из таких правил является своевременное регулирование цен на товар. Итоговая цена кроме себестоимости так же включает и наценку.

*Наценка это часть стоимости продукта, которая составляет разницу между конечной ценой товара и его себестоимостью.*

Факторы определения наценки:

- Налогообложение;
- расходы, связанные с реализацией;
- спрос;
- ситуация на рынке.

Если налогообложение и расходы это преимущественно статическая информация, то спрос и ситуация на рынке всегда динамичны.

Данный алгоритм на основе имеющейся информации позволит скорректировать цену следующего периода на анализе информации предыдущих периодов (новости, цены и спрос предыдущих периодов и т.д.).

## II. ПРИМЕНЕНИЕ НЕЙРОСЕТИ В ЗАДАЧАХ ПРОГНОЗИРОВАНИЯ

Нейросети и инвестиционная деятельность

Нейронные сети привлекательны с интуитивной точки зрения, ибо они основаны на примитивной биологической модели нервных систем. В будущем развитие таких нейробиологи-

ческих моделей может привести к созданию действительно мыслящих компьютеров.

Предсказание финансовых временных рядов - необходимый элемент любой инвестиционной деятельности. Сама идея инвестиций - вложения денег сейчас с целью получения дохода в будущем - основывается на идее прогнозирования будущего. Соответственно, предсказание финансовых временных рядов лежит в основе деятельности всей индустрии инвестиций - всех бирж и небиржевых систем торговли ценными бумагами.

Известно, что 99% всех сделок - спекулятивные, т.е. направлены не на обслуживание реального товарооборота, а заключены с целью извлечения прибыли. Все они основаны на предсказаниях изменения курса участниками сделки. Причем предсказания участников каждой сделки противоположны друг другу. Так что объем спекулятивных операций характеризует степень различий в предсказаниях участников рынка, т.е. степень непредсказуемости финансовых временных рядов.

Нейросетевое моделирование в чистом виде базируется лишь на данных, не привлекая никаких априорных соображений. В этом заключается его преимущество и одновременно – его недостаток. Имеющихся данных может не хватить для обучения, размерность потенциальных входов может оказаться слишком велика.

Применение нейронных сетей в задачах прогнозирования

Задачи прогнозирования особенно важны для практики, в частности, для финансовых приложений, поэтому следует рассмотреть способы применения нейронных сетей в этой области более подробно.

Рассмотрим практическую задачу, ответ в которой очевиден - задачу прогнозирования курса драгоценных металлов на 1 день вперед.

Пусть имеется база данных, содержащая значения курса за последние 628 дней. Необходимо построить прогноз завтрашней цены на основе курсов за последние несколько дней. Прогнозирующая нейронная сеть должна иметь всего

один выход и столько входов, сколько предыдущих значений будет использоваться для прогноза.

Возникает вопрос, прогнозировать абсолютные или относительные значения? В качестве входов и выходов нейросети не следует выбирать сами значения котировок. Действительно значимыми для предсказаний являются изменения котировок. Поскольку эти изменения, как правило, гораздо меньше по амплитуде, чем сами котировки, между последовательными значениями курсов имеется большая корреляция – наиболее вероятное значение курса в следующий момент равно его предыдущему значению. Между тем, для повышения качества обучения следует стремиться к статистической независимости входов, т.е. к отсутствию подобных корреляций.

Поэтому в качестве входных переменных следует выбирать наиболее статистически независимые величины, например, изменения котировок или логарифм относительного приращения. Последний выбор хорош для длительных временных рядов, когда уже заметно влияние инфляции. В этом случае простые разности в разных частях будут иметь различную амплитуду, т.к. фактически измеряются в различных единицах. Напротив, отношения последовательных котировок не зависят от единиц измерения и будут одного масштаба несмотря на инфляционное изменение единиц измерения. В итоге, большая стационарность ряда позволит использовать большую историю и обеспечит лучшее обучение.

#### Входные данные

Вторым и наиболее важным этапом является определение состава входов. Распространенной ошибкой является то, что на вход подается большое количество разнообразной и часто ненужной информации и считается, что нейросеть сама выберет важные показатели.

Действительно, нейросеть умеет отсеивать незначимые входы, но каждый «лишний» вход утяжеляет сеть, затрудняет обучение, а главное - снижает качество прогноза. При большом числе «лишних» входов нейросеть чаще всего делает прогноз по предшествующим показателям. Поэтому необходимо очень аккуратно подходить к добавлению новых входов. На входы нейросетей можно подавать самую разнообразную информацию, как дискретную (различные события) так и непрерывную (цены, индексы, индикаторы). Наиболее значимыми входами являются индикаторы

технического анализа и различные фондовые индексы.

Большой проблемой при определении состава входов является определение глубины «погружения» - размера временного окна, данные из которого подаются на вход нейросети. Возникает вопрос, сколько баров истории подавать на вход, чтобы качественно обучить нейросеть? Эта величина определяется только из собственного опыта и должна быть адекватной прогнозируемому периоду. Например, при дневных прогнозах разумная глубина погружения находится в пределах 5-20 дней.

Необходимо отметить, что многие популярные нейропакеты имеют функцию определения чувствительности по входам, которая рекомендуется разработчиками пакетов как универсальный способ отсеивания «лишних» входов. Хотя эта функция и является полезной при определении состава входов, ее автоматическое использование может принести больше вреда, чем пользы. Это обусловлено тем, что проблема определения чувствительности по входам для многослойных нейронных сетей пока математически не решена, и решается различными эмпирическими способами и алгоритмами. Поэтому рекомендуется пользоваться этой функцией с осторожностью и результаты проверять другими косвенными методами, например, статистическими.

#### ЗАКЛЮЧЕНИЕ

Предлагаемый алгоритм позволит скорректировать цену на прогнозируемый период, на основе графиков предыдущих цен и анализа новостей связанных с этим продуктом. Алгоритм не будет гарантировать предлагаемую им динамику цен со стопроцентной точностью, однако позволит минимизировать риски планирования продаж, предотвратить возможность случайного какого-либо ключевого фактора в процесс анализа и принятия решений.

1. Математические методы принятия решений / В. И. Бодров, Т. Я. Лазарева, Ю. Ф. Мартемьянов // Издательство ТГТУ, 2004
2. Machine Learning and Big Data Analytics Paradigms: Analysis, Applications and Challenges [1st ed. 2021] / A. E. Hassanien, A. Darwish. – Springer, 2021.
3. Осовский С. / Нейронные сети для обработки информации М.: Финансы и статистика, 2004.
4. S. M. Obratsov , A. L. Shimkevich An Zero Order Minimization Algorithm Using a Neural Network // Proceedings of the International Joint Conference on Neural Networks, Washington. July 10-16. – 1999. – V. 1. – 1999. – P. 643-645.

# ИНФОРМАЦИОННАЯ СИСТЕМА УЧЕТА НАУЧНОЙ РАБОТЫ СОТРУДНИКОВ КАФЕДРЫ И ПРОГНОЗИРОВАНИЕ РЕЙТИНГА УНИВЕРСИТЕТА

Плиско И. Г., Ганжин П. С., Ковалев М. А.

Кафедра автоматизированных систем управления, Белорусско-Российский университет  
Могилев, Республика Беларусь

E-mail: ginn\_g\_211@mail.ru, {manutdsinse1878, pavel.gangin}@gmail.com

*В данных материалах описывается информационная система, которая собирает данные с сайта Elibrary и хранит информацию о публикациях преподавателей кафедры АСУ. Кроме того, созданная информационная система позволяет получить информацию о научно-исследовательской работе вузов и их текущих мест в рейтинге Webometrics для будущего прогнозирования позиции в данном рейтинге Белорусско-Российского университета.*

## ВВЕДЕНИЕ

Основным фактором конкурентного успеха университета является его репутация на национальном и международном образовательных рынках. Участие в международных рейтингах – одно из направлений развития Белорусско-Российского университета (БРУ), которое позволяет включить вуз в международную образовательную среду, согласовать образовательные и научно-исследовательские процессы с международными правилами и стандартами в сфере образования.

Webometrics - это крупнейший академический рейтинг высших учебных заведений, который анализирует присутствие университетов в интернете, включая научные публикации. По мнению разработчиков рейтинга, используемые показатели позволяют создать достаточно полную картину деятельности преподавателей и исследователей университетов.[1]

Основная цель создания информационной системы – сбор информации о научно-исследовательской работе ВУЗов и их текущих местах в рейтинге для будущего прогнозирования места в рейтинге Белорусско-Российского университета, а также сбор данные с сайта Elibrary с последующим хранением информации о публикациях преподавателей кафедры АСУ для будущего анализа.

## I. ОРГАНИЗАЦИЯ СБОРА ДАННЫХ

В разработанном приложении использован ряд технологий.

Node – кроссплатформенная среда выполнения для JavaScript, которая работает на серверах. Данная среда исполнения даёт возможность для использования вне контекста браузера (т. е. выполняется непосредственно на компьютере). Таким образом, среда исключает API-интерфейсы JavaScript для браузера и добавляет поддержку более традиционных API-интерфейсов, включая библиотеки HTTP и файловых систем.

PostgreSQL является свободной объектно-реляционной системой управления базами данных. PostgreSQL базируется на языке SQL. и поддерживает многочисленные возможности.

Handlebars – шаблонизатор без логики, который динамически генерирует необходимую HTML-страницу. Это расширение Mustache с несколькими дополнительными функциями.

Для получения данных используются два основных способа получения информации: через API и парсинг данных.

API (Application Programming Interface) представляет комплекс разного рода инструментов, функций, реализованных в виде интерфейса для создания новых приложений, с помощью которого одна программа будет взаимодействовать с другой.

Сайт elibrary (<https://www.elibrary.ru>) имеет свой API, доступ к которому можно получить по подписке SINCE INDEX. [2] Белорусско-Российский университет имеет такую подписку, но она даёт ограниченный доступ, не дающий возможность реализовать интересующую задачу. В связи с чем был выбран вариант парсинга данных.

Парсинг (Parsing) – это принятое в программировании определение синтаксического анализа. Для этого выстраивается математическая модель сравнения лексем с формальной грамматикой, описанная одним из языков программирования (в нашем случае с помощью Node. Js). Вне зависимости от использованного языка программирования, на котором написан парсер, алгоритм его действия сохраняется одинаковым:

- выход в интернет, получение доступа к коду веб-ресурса;
- чтение, извлечение и обработка данных;
- представление извлеченных данных в требуемом виде. Созданная информационная система представляет собой несколько объединенных вместе страниц. При входе пользователь видит состав кафедры с демонстрацией наукометрических показателей авторов.

## II. ПРОГНОЗИРОВАНИЕ РЕЙТИНГА УНИВЕРСИТЕТА

Для построения адекватной модели нужно проверить, чтобы ее коэффициент детерминации был выше 0,7. В противном случае модель отклоняется из-за недостаточной статистической связи между исследуемыми величинами.

Предварительные исследования показывают, что между количеством публикаций и количеством цитирований существует тесная статистическая связь, оцениваемая коэффициентом детерминации, большим 0,75. Поэтому построение прогнозной модели выполняется на основе числа публикаций в РИНЦ. Число публикаций в РИНЦ – это публикации, в которых сотрудник организации является автором или соавтором. В них не учитываются работы, где он является редактором, составителем, переводчиком и т.д. Также не учитываются статьи в информационных, научно-популярных изданиях, а также журналах, исключенных из РИНЦ.

При переходе на страницу «Анализ публикационной активности организации» (сайт [Elibrug.ru](http://elib.rug.ru)) для каждого университета, занимающего первые 10 мест в рейтинге, и Белорусско-Российского университета, получают данные о числе публикаций в РИНЦ. По этой информации и данных о месте в рейтинге Webometrics, взятых с сайта Webometrics для этих же ВУЗов, строятся линии тренда.

Информационная система прогнозирует данные на основе пяти моделей: линейная, логарифмическая, экспоненциальная, степенная и полиномиальная.

На главной панели находится панель вкладок, по умолчанию открыта вкладка на которой выводится полная информация о модели. На вкладке «Подробнее» находится информация с пояснениями о выводимых данных. За ней идет секция прогноз, в котором выводится информация о прогнозировании рейтинга БРУ для вводимого количества публикаций. Если коэффициент детерминации меньше 0,7, то прогноз осуществляться не будет, так как недостаточно выражена зависимость.

## III. РЕКОМЕНДАЦИИ ПО ПОВЫШЕНИЮ РЕЙТИНГА

Для того чтобы занять достойное место в рейтинге, ВУЗу необходимо вести работу по следующим направлениям:

- Создание научного контента, поддержке пишущих преподавателей и продвижении материалов в сети. Чем больше публикаций создается внутри университета, тем выше его рейтинг в Webometrics.

- Работа с сайтом университета. К вузовскому сайту нужно относиться как к виртуальному, полноценному представительству научного массива результатов исследований в мировой сети Интернет. Для этого надо строго структурировать контент на сайте в соответствии с целевыми аудиториями. В результате сайт станет более удобным в использовании, что облегчит Webometrics поиск научного и другого контента ВУЗа.
- Организация и проведение тематических семинаров и конференций. Университет должен заниматься наращиванием контента. Для этого могут использоваться материалы конференций, семинаров, «круглых столов», публикации образовательных программ, монографий, препринтов, диссертаций, научных отчетов.
- Создание и публикация препринтов на иностранных языках на сайте университета.
- Разработка мобильных приложений для платформ iOS и Android (количество посетителей в день – 6 млн и 20 млн соответственно) в целях продвижения вузовской научной и образовательной продукции.
- Создание персональных страниц, блогов руководителей, преподавателей и студентов университета. Это важнейший ресурс для повышения научного рейтинга любого вуза. Однако следует заметить, что блоги нужно открывать не на сторонних площадках, а на сервере университета, где располагается его сайт.

## ЗАКЛЮЧЕНИЕ

Преимуществами созданной информационной системы являются доступность, целенаправленное выполнение поставленных задач и простота в использовании программным продуктом.

Её пользователями могут быть сотрудники кафедры и её руководитель, которые будут применять приложение для отслеживания деятельности сотрудников и её анализа, что поможет увеличить научные показатели кафедры и университета в целом, в следствии чего сможет вырасти позиция университета в международных рейтингах.

1. WEBOMETRICS / Ranking Web of Universities [Электронный ресурс]: официальный сайт. – Режим доступа: <http://www.webometrics.info/en> .– Дата доступа: 14.04.2021.
2. Научная электронная библиотека eLIBRARY.RU [Электронный ресурс] – Режим доступа: <https://www.elibrary.ru/defaultx.asp> .– Дата доступа: 28.09.2021.

# ОЦЕНКА УСТОЙЧИВОСТИ КРАТЧАЙШИХ ПУТЕЙ НА ДИНАМИЧЕСКИ ОПРЕДЕЛЯЕМЫХ ГРАФАХ

Ревотюк М. П., Бебех А. В., Хаджинова Н. В.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: {rmp, kafitas}@bsuir.by

*Предлагаются ускоренные процедуры оценки устойчивости решения задачи поиска кратчайших путей на динамически нагруженных орграфах, когда порядок порождаемых деревьев путей существенно меньше порядка графа. Под устойчивостью решения понимается его нечувствительность к независимым аддитивным возмущениям весов ребер графа. На основе метода бутстрэппинга показана возможность снижения сложности оценки устойчивости до линейной зависимости от объема сканируемого пространства.*

## I. ПОСТАНОВКА ЗАДАЧИ

Известно, что для поиска кратчайших путей на нагруженном ориентированном графе  $G(N, A)$ , где  $N$  – множество вершин,  $A$  – множество дуг с весовой функцией  $W : A \rightarrow R^+$ , лучшим является алгоритм Дейкстры [1]. Процесс построения дерева кратчайших путей имеет волновой характер до исчерпания возможности его развития из исходной вершины. В этом случае для каждой вершины  $x$  требуется представление множества непосредственно достижимых смежных вершин  $x'$ ,  $x' = \{y | w(x, y) \geq 0\}$ , где  $w(x, y)$  – вес дуги  $x \rightarrow y$ ,  $y \in N$ .

На практике возникает потребность поиска путей на графах с изменяемой структурой, когда описание графа задано легко модифицируемым списком дуг. Реляционная модель графа и пространства поиска решения – основа эффективной реализации поиска дерева путей методом бутстрэппинга [2]. Такой метод отличается ограничением потребности в памяти количеством фактически просматриваемых вершин без оценки предельных значений  $|N|$ , а также возможностью реализации открытых для расширения систем.

Анализ известных алгоритмов оценки устойчивости кратчайших путей [3,4], сложность которых  $O(|A| \log(|N|))$ , показывает возможность их улучшения путем обработки фактически просматриваемых элементов графа.

Предмет обсуждения – развитие и обобщение приемов реализации метода бутстрэппинга [2] в задачах поиска и оценки устойчивости кратчайших путей на графах с переменной структурой.

## II. БАЗОВЫЙ АЛГОРИТМ

Представим переменные состояния процесса построения дерева путей элементами отношения  $T(x, d, p)$ , где  $x$  – номер узла дерева (уникальный элемент),  $d$  – расстояние от корня  $s$  до узла  $x$ ,  $p$  – номер предшествующего узла кратчайшего пути в узел  $x$ . По определению,  $L(s)$  – проекция отношения  $T(x, d, p)$  по атрибуту  $x$ .

Очевидно, что состояние процесса построения дерева на любом этапе  $k$  представлено тройкой  $(x_k, d(x_k), p(x_k))$ . Начальное состояние процесса построения дерева с корнем  $s$  соответствует тройке  $(s, 0, s)$ . Условие завершения процесса –  $(L^k = \emptyset)$  или  $(x_k = t)$ , если  $t$  – конечная вершина пути. В любом случае  $k \leq |L(s)|$  после окончания поиска пути  $s \rightarrow t$ . Нумерация состояний в явном виде не потребуется, если в операции выборки очередного элемента  $x_k$  из  $L^k$ , обозначаемой далее как  $L^k \ominus x_k$ , контролировать условие завершения процесса.

Результат операции  $L^k \ominus x_k$  – истинность условия  $(L^k \neq \emptyset)$ , новое значение  $x_k$  в случае  $(L^k \neq \emptyset)$  и  $k \leftarrow k + 1$ . Так как по определению  $d(x_k) \leq d(x_{k+1})$ , то операция ' $\ominus$ ' требует упорядочения элементов  $L(s)$  по ключу  $d(x) \odot x$ , где символ ' $\odot$ ' обозначает операцию конкатенации.

Операцию расширения дерева кратчайших путей обозначим символом ' $\oplus$ '. Результат операции  $L^k \oplus x_k$  – истина, если  $x_k \notin L(s)$  и выполнено  $L^k \leftarrow L^{k-1} \cup \{x_k\}$ . Ее реализация требует упорядочения элементов  $L(s)$  по ключу  $x$ .

Алгоритм Дейкстры поиска кратчайшего пути  $s \rightarrow t$  в терминах определенных выше операций имеет вид:

```
function sp1(s, t) begin
 d(s) = 0; p(s) = s; L* = {s};
 while (L* \ominus x) \wedge (x \neq t) do
 L \oplus x;
```

Здесь этап расширения дерева кратчайших путей из вершины  $x$  отражает функция

```
function relax(x) begin
 foreach (y \in x') do
 r = d(x) + w(x, y);
 if (L* \oplus y) \vee (d(y) > r) then
 d(y) = r; p(y) = x;
```

Любая вершина  $x$ , к которой применялась операция  $relax(x)$ , становится узлом дерева кратчайших путей с постоянными значениями  $p(x)$  и  $d(x)$ .

Дерево кратчайших путей – связный граф по определению. Если  $s$  – исходная вершина,

а  $x$  – произвольный узел или лист дерева путей,  $s, x \in N$ , то после завершения поиска последовательность посещаемых вершин обратного пути  $\{x, p(x), p(p(x)), \dots, s\}$  представляет искомый кратчайший путь. Расширение дерева кратчайших путей происходит только из некоторого листа без пометки, а листья из множества  $L(s) \setminus L^*$  представляют лишь исторический интерес и становятся пассивными. Обычно на практике  $|L(s)| \ll |N|$ , что объясняет привлекательность схемы бутстрэппинга.

### III. АЛГОРИТМ ОЦЕНКИ УСТОЙЧИВОСТИ РЕШЕНИЯ

Рассмотрим вершину  $y$ , для которой на кратчайшем пути  $s \rightarrow y, s, y \in N$ . постоянно фиксируется расстояние  $d(y)$  и  $p(y)$ . Изменение структуры такого пути возможно после изменения весов любой из входных дуг вершины  $y - \{(x, y), x \in 'y\}$ . Здесь  $'y = \{x | w(x, y) \geq 0\}$  – множество входных вершин дуг в вершину  $y$ , где  $x, y \in N$ .

Выходные дуги вершин из множества  $'y$  в вершину  $y$  следует разделить на три категории: последняя дуга кратчайшего пути в вершину  $y$  (дуга  $(p(y) \rightarrow y)$ ;

дуги вне дерева кратчайших путей из вершин, принадлежащих такому дереву;

дуги из вершин, не рассматриваемых до момента постоянной пометки вершины  $y$ .

Дуги вне дерева кратчайших путей из вершин, принадлежащих такому дереву, можно выделить на графе в момент ветвления альтернатив развития дерева путей из каждой вершины из множества  $'y$ . Множество входных дуг вне дерева кратчайших путей, появляющихся после построения дерева, остаются без пометки вхождения соответствующих дугам инцидентных вершин в список  $L^*$ .

Обозначим интервал допустимого изменения веса дуги как  $[w_{min}(x, y), w_{max}(x, y)]$  для каждой дуги  $x \rightarrow y, x, y \in L(s)$ . Далее, для удобства различения принадлежности интервалов, пометим верхним индексом ‘-’ границы весов дуг дерева кратчайших путей, а индексом ‘+’ – дуг вне дерева кратчайших путей.

Интервал  $[w_{min}^+(x, y), w_{max}^+(x, y)]$  допустимого изменения весов дуг  $x \rightarrow y$ , не входящих в дерево кратчайших путей, определяется выражениями

$$\begin{aligned} w_{min}^+(x, y) &= d(y) - d(x), \\ w_{max}^+(x, y) &= \infty, (x \in 'y) \wedge (x \neq p(y)). \end{aligned}$$

Интервал  $(w_{min}^-(x, y), w_{max}^-(x, y)]$  допустимого изменения весов дуг  $x \rightarrow y$ , входящих в дерево кратчайших путей, определяется выражениями

$$\begin{aligned} w_{min}^-(x, y) &= 0; \\ w_{max}^-(x, y) &= w(x, y) + \delta(y) - d(y), \\ \delta(y) &= \min\{d(v) + w(v, y) | v \in 'y\}. \end{aligned}$$

Последние выражения определены относительно момента постоянной пометки очередной вершины  $x_k$  растущего дерева кратчайших путей (момента вызова операции  $relax(x_k)$ ).

Нетрудно заметить, что процессы вычисления границ интервалов допустимого изменения весов дуг определены лишь на входных дугах очередной вершины. Такие процессы могут быть совмещены с целью однократного сканирования списка входных дуг вершины  $y$ :

```
function spr(y) begin
 $\delta = \infty; q = p(y);$
 foreach $(x \in 'y) \wedge (x \in L(s) \setminus L^*)$ do
 $\delta = \min(\delta, d(x) + w(x, y));$
 $w_{min}^+(x, y) = d(y) - d(x);$
 $w_{max}^+(x, y) = \infty;$
 $w_{min}^-(q, y) = 0;$
 $w_{max}^-(q, y) = w(q, y) + \delta - d(y);$
```

Важно отметить, что условие  $(x \in L(s) \setminus L^*)$  позволяет выделить лишь действительно влияющие на устойчивость пути дуги в момент постоянной пометки вершины  $y$ .

Очевидно, что оценка рассмотренных интервалов устойчивости может быть отложена. Этим можно воспользоваться для исключения холостых вычислений неустребованных данных. Например, возможны следующие варианты выполнения оценки интервалов устойчивости:

после построения дерева кратчайших путей для всех его дуг;

на каждой итерации построения дерева кратчайших путей;

после построения дерева кратчайших путей для смежных дуг конкретных кратчайших путей.

### ЗАКЛЮЧЕНИЕ

Таким образом, алгоритм оценки устойчивости кратчайших путей на неоднородных динамически определяемых графах представлен в форме, согласованной с возможностями максимального сокращения вычислительной сложности методом бутстрэппинга.

### СПИСОК ЛИТЕРАТУРЫ

1. Ferone, D. Shortest paths on dynamic graphs: a survey/D. Ferone [et al.]/Pesquisa Operacional, 2017. – Vol. 37, iss. 3. – P. 487–508.
2. Ревотюк, М. П. Встречный поиск кратчайших путей на больших динамических графах методом бутстрэппинга/М.П. Ревотюк, Н.В. Хаджинова //BIG DATA and Advanced Analytics: сб. материалов VI Междунар. науч.-практ. конф., Минск, 20–21 мая 2020 г. В 2 ч. Ч. 1. – Минск: Бестпринт, 2020. – С. 324–331.
3. Shier, D. R., Witzgall C. Arc tolerances in shortest path and network flow problem// Networks. –1980. – Vol. 10. – P. 277–291.
4. Tarjan, R. E. Sensitivity analysis in minimal spanning trees and shortest path trees//Inf. Proc. Letters. – 1982. – Vol. 14, iss. 1. – P. 30–33.

# МАШИННОЕ РАСПОЗНАВАНИЕ ЭМОЦИЙ ПО ГОЛОСУ

Серебряная Л. В., Брановицкий А. А.

Кафедра программного обеспечения информационных технологий,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: l\_silver@mail.ru

*Работа посвящена распознаванию и классификации эмоций человека по голосу с помощью машинного обучения. В ней выполнен анализ программных продуктов, методов и моделей, предназначенных для распознавания эмоций по речи. Определены признаки, извлекаемые из речевого сигнала, оказывающие основное влияние на точность распознавания эмоций системой-классификатором. Предложен алгоритм нахождения этих характеристик звукового сигнала. Выбрана архитектура искусственной нейронной сети для распознавания эмоций по речи. Классифицированы эмоции по звуковому сигналу, а также оценено качество полученных результатов.*

## ВВЕДЕНИЕ

Объектом рассмотрения в работе выбран голос, поскольку он является одним из важнейших каналов для идентификации эмоционального состояния человека. На современном этапе развития информационных технологий разработка методов и систем автоматического распознавания эмоций по речевому сигналу с помощью аппаратно-программных средств является актуальной задачей, связанной с применением неинвазивных средств диагностики и мониторинга психофизиологического состояния человека [1]. На основании визуальных данных можно с высокой точностью предсказывать знак эмоции, но при определении интенсивности предпочтительнее использовать речевые сигналы. В настоящее время создано большое количество систем распознавания эмоций на основе мимики, обладающих довольно высокой точностью получаемых результатов, а систем распознавания эмоций по голосу – намного меньше.

## I. ОСОБЕННОСТИ РАСПОЗНАВАНИЯ ЭМОЦИЙ ПО ГОЛОСУ

Для конструктивного решения задачи автоматического распознавания эмоций по речи необходимо количественно охарактеризовать речевой сигнал и выделить существенные параметры, отвечающие за эмоции человека. Обычно для анализа звука используется ряд его признаков, которые подаются на вход системы-классификатора.

Анализ программ-прототипов (Empath, EMOSpeech), предназначенных для распознавания эмоций по речи, позволил сделать выводы о том, что рассматривается довольно ограниченный набор эмоций (от 4 до 6). Кроме того, программные средства, характеризующиеся высокой точностью, имеют закрытый исходный код, что создает определённые трудности по их дальнейшему усовершенствованию.

Поскольку сложно заранее сформулировать формальные правила, по которым можно определить эмоциональное состояние человека по ре-

чевому сигналу, то использование машинного обучения для создания системы распознавания эмоций по речевому сигналу оправдано.

Метод опорных векторов и деревья решений часто используются в несложных системах распознавания эмоций по речевому сигналу. Эти простые модели машинного обучения позволяют быстро получить первые результаты классификации, понять специфику задачи и оценить, для каких объектов качество прогнозирования высокое, а для каких – низкое. Наиболее перспективными моделями для рассматриваемых задач признаны искусственные нейронные сети (ИНС), поскольку они устойчивы к шумам во входных данных, показывают высокую степень классификации и позволяют настраивать параметры модели для каждой конкретной ситуации.

В работе системы распознавания эмоций по речевому сигналу выделяют четыре основных этапа:

- предварительная обработка сигнала;
- выделение характерных особенностей речевого сигнала;
- предобработка особенностей речевого сигнала;
- классификация.

Для повышения быстродействия проектируемой системы в качестве алгоритма выделения спектральных составляющих цифрового сигнала удачнее всего использовать быстрое преобразование Фурье (БПФ). Для снижения спектра частотного отклика на коротких временных промежутках лучшими характеристиками обладает окно Хемминга. Качество классификации напрямую зависит от характерных признаков речевого сигнала. Основными из которых являются: *melfrequency cepstral coefficients* (мел-кепстральные коэффициенты), MFCC-коэффициенты и форманты. Преимущества использования MFCC-коэффициентов объясняется следующими факторами:

1. Используется спектр сигнала, то есть разложение по базису ортогональных синусо-



- идальных и косинусоидальных функций, что позволяет учитывать волновую «природу» сигнала при дальнейшем анализе.
2. Спектр проецируется на специальную мел-шкалу, позволяя выделить наиболее значимые для восприятия человеком частоты.
  3. Количество вычисляемых коэффициентов может быть ограничено любым значением, что позволяет «сжать» фрейм сигнала и, как следствие, уменьшить количество обрабатываемой информации.

Однако для достижения высокой точности распознавания эмоций по речевому сигналу одних этих параметров не всегда достаточно. Комбинирование MFCC-коэффициентов и формант с другими характерными особенностями речевого сигнала перед подачей на вход классификатора позволяет достигать высокой точности классификации эмоций [2, 3].

В связи с тем, что MFCC-коэффициенты оказывают наибольшее влияние на точность работы классификатора, рассмотрим алгоритм вычисления MFCC-коэффициентов.

1. Считать данные из аудиофайла.
2. Выполнить частотную фильтрацию сигналов.
3. Разбить данные на фреймы.
4. Применить окно Хэмминга к фреймам.
5. Выполнить БПФ.
6. Наложить на данные банк треугольных мел-фильтров.
7. Выполнить логарифмирование энергии частотной области.
8. Применить дискретное косинусное преобразование Фурье.

MFCC-коэффициенты вычисляются по следующей формуле

$$C_i(n) = \sum_{k=0}^{K-1} L(k) \cos\left(\frac{\pi * n}{K} \left(k + \frac{1}{2}\right)\right),$$

где  $k=0, 1, \dots, K-1$  — индекс фильтра,  $K$  — количество фильтров,  $n=0, 1, \dots, N-1$  — индекс отсчета,  $N$  — количество отсчетов,  $L(k)$  — энергия окна.

Частотная фильтрация предполагает выделение низких частот, поскольку к ним наиболее чувствителен человеческий слух. Разбиение на фреймы заключается в разделении сигнала на фрагменты длительностью 30-40 мс с 50% перекрытием. Дискретное косинусное преобразование Фурье выполняется для уменьшения размерности вектора MFCC-коэффициентов.

## II. РАСПОЗНАВАНИЕ ЭМОЦИЙ С ПОМОЩЬЮ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ

Для достижения высокой точности распознавания эмоций по речевому сигналу необходимо определиться с архитектурой ИНС. В работе

был выбран многослойный перцептрон, состоящий из 40 нейронов во входном слое, 24 нейронов в скрытом слое и 8 — в выходном [4]. Исходными данными для обучения многослойного перцептрона явились 40 нормализованных MFCC-коэффициентов, выходными данными — одна из 8 эмоций, аудиофайлы которых представлены в библиотеке RAVDESS. В качестве функции активации взята  $\text{relu}$ . Точность распознавания эмоций в данной конфигурации составляет порядка 70%. Дальнейшие эксперименты с изменением архитектуры ИНС при неизменных исходных данных позволили достичь точности порядка 90% для пятислойного перцептрона с размерностью слоёв: 40-300-150-25-8.

В ходе экспериментальных исследований было установлено, что лучшим классификатором среди рассмотренных для системы распознавания эмоций по речевому сигналу является многослойный перцептрон. Вычисление MFCC-коэффициентов речевого сигнала позволяет достигать высокой точности определения эмоционального окраса речи при разумных вычислительных затратах. Поэтому в качестве метода нахождения нужных характеристик речевого сигнала перед подачей на вход классификатору был выбран алгоритм вычисления мел-кепстральных коэффициентов.

Наиболее легко классифицируемыми эмоциями оказались спокойствие и нейтральная эмоция, хотя размер тестовой выборки для нейтральной эмоции меньше в 2 раза по сравнению с тестовыми выборками для всех остальных эмоций.

## ЗАКЛЮЧЕНИЕ

Высокую точность (свыше 90%) классификации эмоций по речевому сигналу удалось достичь за счёт разработки методов аугментации речевых данных и увеличения исходной тестовой выборки в 5 раз. Сравнение результатов распознавания эмоций созданной системой с существующими аналогами позволило сделать выводы о том, что разработанная система обладает точностью распознавания эмоций, сопоставимой с аналогами, а также не уступает им в быстродействии.

1. Wanare, A. Human Emotion Recognition From Speech / A. Wanare, S. Dandare // Journal of Engineering Research and Applications. — 2014. — vol. 4, № 7. — p. 74-78.
2. Ingale A. Speech Emotion Recognition / A. Ingale, D. Chaudhari // International Journal of Soft Computing and Engineering (IJSCE).— March 2012. — p. 235-238.
3. Utane A. Emotion Recognition through Speech / A. Utane, S. Nalbalwar // International Journal of Applied Information Systems (IJ AIS). — 2013. — p. 5-8.
4. Николенко, С. Глубокое обучение. Погружение в мир нейронных сетей / С. Николенко, А. Кадури, Е. Архангельская. — Спб.: Питер, 2019. — 480 с.

# ОСОБЕННОСТИ ПОДГОТОВКИ СПЕЦИАЛИСТОВ В ОБЛАСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Тиханович Т. В., Боброва Т. С.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {tihanovich, t.bobrova}@bsuir.by

*Рассматриваются особенности подготовки специалистов в области информационных технологий с учетом различных подходов к организации учебного процесса. Отмечается необходимость формирования специалиста, востребованного на рынке труда, с учетом современных тенденций развития общества.*

## ВВЕДЕНИЕ

Современное развитие экономики и социальной среды непосредственно зависит от уровня обучения и воспитания будущих специалистов, которые должны обладать соответствующими компетенциями, позволяющими быстро адаптироваться в новых условиях и находить наиболее правильное и креативное решение для различных задач профессиональной деятельности.

На сегодняшний день особенно востребованы квалифицированные работники в отраслях, связанных с электронной обработкой большого объема данных и разработкой соответствующего программного обеспечения, поддерживающего автоматизацию различных аспектов деятельности.

До недавнего времени обучение студентов в Республике Беларусь базировалось на получении фундаментальных знаний и подготовке специалиста широкого профиля, что привело к дефициту практико-ориентированных специалистов. Ведущая роль в образовательных стандартах нового поколения отводится компетенциям. В общем случае под компетенциями понимается способность применять знания, умения, навыки, личностные качества для успешной деятельности в различных ситуациях, связанных с профессиональной и социальной деятельностью. Таким образом, перед сферой образования в целом и, в частности, перед сферой высшего образования стоит задача подготовки востребованных специалистов с учетом компетентного подхода.

## I. ТРЕБОВАНИЯ К КВАЛИФИКАЦИИ СОВРЕМЕННЫХ СПЕЦИАЛИСТОВ ИТ-ПРОФИЛЯ

В образовательном стандарте для специальностей ИТ-профиля формулируются общие цели подготовки специалиста следующим образом. Во-первых, необходимо сформировать и развить социально-профессиональную компетентность, которая позволяет сочетать академические, профессиональные, социально-личностные компетенции для решения задач в сфере профессиональной и социальной деятельности. Во-вторых, требуется сформировать навыки профессиональной деятельности, которые включа-

ют в себя умение ставить задачи, вырабатывать и принимать решения с учетом их социальных, экологических и экономических последствий, а также планировать и организовывать работу коллектива. В-третьих, немаловажным является и формирование навыков исследовательской работы, которая предполагает планирование и проведение научного эксперимента, умение проводить научный анализ полученных результатов, осуществлять творческое применение научных достижений в области программного обеспечения информационных технологий.

Исходя из целей подготовки, а также учитывая виды и задачи профессиональной деятельности будущих специалистов ИТ-профиля, в образовательном стандарте отмечены следующие группы основных компетенций: академические, социально-личностные и профессиональные. Академические компетенции включают умения работать самостоятельно и постоянно повышать свой профессиональный уровень, применять полученные базовые научно-теоретические знания для решения научных и практических задач в области создания и совершенствования программного обеспечения информационных технологий, формулировать и выдвигать новые идеи. Социально-личностные компетенции связаны с тем, что будущий специалист должен иметь высокую гражданственность и патриотизм, знать права и соблюдать обязанности гражданина, иметь способность находить правильные решения в условиях чрезвычайных ситуаций, обладать способностью к межличностным коммуникациям, уметь работать в коллективе[1].

Основой квалификации современных специалистов ИТ-профиля являются профессиональные компетенции, предполагающие, что выпускники должны обладать знаниями и умениями формулировать проблемы и решать задачи, разрабатывать планы и обеспечивать их выполнение в избранной сфере профессиональной деятельности.

Таким образом, специалист, получивший образование в области ИТ-технологий, должен обладать профессиональными компетенциями в

проектно-производственной, эксплуатационной, организационно-управленческой, инновационной сферах, а также в научно-исследовательской и образовательной областях.

## II. ОСНОВНЫЕ ПРОБЛЕМЫ ПОДГОТОВКИ СОВРЕМЕННЫХ ИТ-СПЕЦИАЛИСТОВ

В течение последних лет формировались задачи, предполагающие коллективный творческий подход для достижения поставленных целей в области подготовки ИТ-специалистов. При формировании групп всегда учитываются: психологическая совместимость студентов (желание студентов работать вместе), успеваемость, а также – пожелания по выбору предметной области решаемой задачи и конкретных средств ее реализации и т. д. Кроме того, при проведении учебных занятий активно используются современные образовательные технологии, поддерживающие все стадии работы над индивидуальными проектами, обеспечивающие мультимедийное чтение лекций, планирование проектных работ, осуществление поддержки коллективной работы над проектами, проведение интерактивных индивидуальных консультаций и т.д. С учетом вышеизложенного, научный аспект присутствует практически с самого начала взаимодействия преподавателя и студентов, что выражается, прежде всего, в индивидуальном подходе в определении сложности и новизны разрабатываемого проекта, формулировки задачи исследования для обучающихся, а также – в реальной востребованности проекта для дальнейшего привлечения студентов к выполнению реальных научных исследований, связанных с инновационными разработками.

Программные продукты, с которыми приходится работать студентам, связанным с информационными технологиями, периодически меняются. Подготовка специалистов в области информационных технологий должна обеспечивать актуальность получаемых ими знаний, поэтому будущие специалист должны иметь опыт работы с последними версиями программных продуктов, что бывает достаточно сложно обеспечить на практике. При подготовке учебно-методической документации значительная доля времени в работе преподавателя должна отводиться не совершенствованию оформления учебно-методического материала, а изменению его содержания, связанного с изменением версии программного продукта, что требует значительных временных затрат. Наличие требований к учебно-методическим комплексам в системе высшего образования приводит к тому, что преподаваемый материал приходится оформлять в сжатые сроки, что может приводить либо к подаче студентам устаревшего материала, либо снижению качества документирования преподаваемого курса. Последнее является существенным

недостатком, так как значительная часть времени студентов приходится на самостоятельное изучение преподаваемого курса (или самостоятельное изучение некоторых вопросов, либо разделов курса). Еще одной проблемой является то, что старые и новые версии программ могут использовать различные несовместимые между собой форматы хранения данных. Причем необязательно, что более новая программа обладает возможностью чтения и редактирования устаревших форматов. Во многих случаях форматы хранения данных претерпевают значительные изменения в связи с их низкой эффективностью, и обеспечить совместимость устаревших форматов и новых не представляется возможным. В связи с этим, перед студентами встает вопрос необходимости изучения фактически всех форматов хранения данных, что представляется практически невозможной задачей. [3]. Также невозможно выучить все известные на данный момент языки программирования, разобраться со всеми существующими средами программирования. Появление некоторых новых языков приводит к тому, что некоторые из уже известных языков перестают быть актуальными, теряют свою привлекательность, перестают использоваться. Поэтому еще одной особенностью подготовки современного специалиста в области информационных технологий является то, что заранее неизвестно, будет ли востребован тот язык программирования, который изучается в курсе информатики к моменту трудоустройства или нет.

## ЗАКЛЮЧЕНИЕ

Постоянно нарастающий темп изменений, происходящих в области информационных технологий, требует столь же быстрых изменений в области получения знаний, что практически не является возможным. В связи с этим особое внимание следует уделить процессу самоподготовки, эффективной организации процесса самообразования. Необходимо регулярно проводить исследования в области рынка труда и вносить коррективы в последовательность и объем самоподготовки по тому или иному направлению подготовки. Это позволит обеспечить достаточный уровень знаний в узкой области и в тоже время достаточной гибкостью, чтобы быстро переориентироваться на другие технологии, в случае смены вида деятельности или устаревания используемых информационных продуктов.

1. Изосимова, Т. Н. Компетентностный подход как гарантия качества подготовки современных специалистов в области ИТ-технологий / Т.Н. Изосимова, Л. В. Рудикова // Науч. тр. Акад. упр. при Президенте Респ. Беларусь. Вып. 1. Минск : Акад. упр. при Президенте Респ. Беларусь, 2014. С. 202–209.
2. Сидоров, А. В. Особенности подготовки специалистов в области информационных технологий / А. В. Сидоров. — Текст : непосредственный // Молодой ученый. — 2017. — № 21.1 (155.1). — С. 49–50.

# АЛГОРИТМ ЗАГРУЗКИ МНОГОМЕРНЫХ ДАННЫХ В ДЕНОРМАЛИЗОВАННОЕ ХРАНИЛИЩЕ

Христофорова А. А., Гуринович А. Б.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь  
E-mail: nastafirik@gmail.com, gurinovich@bsuir.by

*Работа посвящена обеспечению обработки многомерных данных в денормализованное хранилище данных. Описано основное назначение OLAP-систем. Проанализированы проблемы при создании хранилищ данных. Детально освещается процесс загрузки данных в хранилище, с использованием технологии ETL, с последующим извлечением, преобразованием и загрузкой данных.*

## ВВЕДЕНИЕ

В современном мире компьютерные сети и вычислительные системы позволяют анализировать и обрабатывать большие массивы данных. Большой объем информации усложняет поиск решений, но дает возможность получить намного более точный расчет и последующий анализ полученных решений. Для этого существуют информационные системы, называемые системами поддержки принятия решений. В ходе работы будет рассмотрена эффективность использования таких подсистем. Прежде, чем приступить к анализу данных, необходимо выполнить обработку данных, цель которой — доведение данных до определенного уровня качества и информативности, а также организовать их интегрированное хранение в структурах, обеспечивающих их целостность, непротиворечивость, высокую скорость и гибкость выполнения аналитических запросов. Исследовался инструмент для (статистической) обработки данных, в основе которого лежит процесс ETL. Данный процесс представляет собой комплекс операций, реализующих процесс переноса первичных данных из различных источников в аналитическое приложение или поддерживающее его хранилище данных. Является составной частью этапа консолидации данных в анализе данных.

### I. ОСНОВНЫЕ ЗАДАЧИ И ПРОБЛЕМАТИКА ПОСТРОЕНИЯ ХРАНИЛИЩ ДАННЫХ

Технология хранилищ данных предназначена для хранения и анализа больших объемов данных с целью дальнейшего обнаружения в них скрытых закономерностей. Термин «OLAP» неразрывно связан с термином «хранилище данных». Основное назначение OLAP-систем — поддержка аналитической деятельности, произвольных запросов пользователей-аналитиков. Целью OLAP-анализа является проверка возникающих гипотез. Основная проблематика при создании хранилищ данных заключается в следующем:

- интеграция разнородных данных;
- эффективное хранение и обработка больших объемов данных;

- организация многоуровневых справочников метаданных;
- обеспечение информационной безопасности хранилища данных.

Условия высокой конкуренции и растущей динамики внешней среды диктуют повышенные требования к системам управления предприятия. Развитие теории и практики управления сопровождалось появлением новых методов, технологий и моделей, ориентированных на повышение эффективности деятельности. Методы и модели в свою очередь способствовали появлению аналитических систем.

### II. ОБЗОР ПРОЦЕССА ЗАГРУЗКИ ДАННЫХ В ХРАНИЛИЩЕ

Извлечение данных из разнотипных источников и перенос их в хранилище данных с целью дальнейшей аналитической обработки связаны с рядом проблем:

- исходные данные расположены в источниках самых разнообразных типов и форматов, созданных в различных приложениях, и, кроме того, могут использовать различную кодировку;
- данные в источниках обычно излишне детализированы, тогда как для решения задач анализа в большинстве случаев требуются обобщенные данные;
- исходные данные, как правило, являются «грязными» (отсутствующие, неточные или бесполезные данные с точки зрения практического применения), что мешает их корректному анализу.

Поэтому для переноса исходных данных из различных источников в хранилище следует использовать специальный инструментарий, который должен извлекать данные из источников различного формата, преобразовывать их в единый формат, а при необходимости — производить очистку данных от факторов, мешающих корректно выполнять их аналитическую обработку.

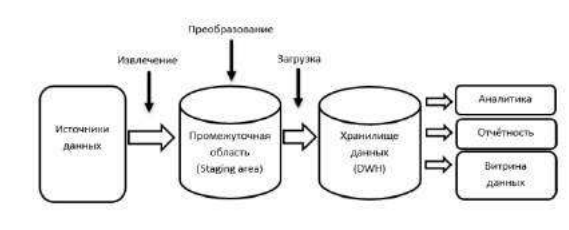


Рис. 1 – Процесс работы ETL-систем

На этапе извлечения данные извлекаются из одного или нескольких источников и подготавливаются к преобразованию. Для корректного представления данных после их загрузки в хранилище из источников должны извлекаться не только сами данные, но и информация, описывающая их структуру, из которой будут сформированы метаданные для хранилища. Процесс преобразования данных источников как правило включает в себя:

- преобразование типов данных;
- преобразования, связанные с нормализацией или денормализацией схемы данных;
- преобразования ключей, связанные с обеспечением соответствия бизнес-ключей суррогатным ключам;
- преобразования, связанные с обеспечением качества данных. Основная сложность на данном этапе заключается в получении требований о том, как именно данные должны быть преобразованы. Заключительный этап подразумевает быструю загрузку данных в хранилище данных.

Существует ряд особенностей данного этапа:

- загрузка данных, основанная на использовании команд обновления SQL, является медленной. Поэтому загрузка с помощью встроенных в СУБД средств импорта/экспорта является предпочтительной;
- индексы таблиц загружаются медленно. Во многих случаях целесообразно удалить индекс и построить его заново;
- следует максимально использовать параллелизм при загрузке данных.

Следует заметить, что при загрузке данных должна быть гарантирована ссылочная целост-

ность данных, а агрегаты должны быть построены и загружены одновременно с подробными данными.

### III. ЗАКЛЮЧЕНИЕ

Исследовалась актуальная область современных информационных технологий — системы анализа данных. Проанализирован основной инструмент аналитической обработки информации - OLAP - технологии. OLAP-технологии — это мощный инструмент обработки данных в реальном времени. Классифицированы основные проблемы создания хранилищ данных. Показано, что эффективным инструментом обработки и анализа данных задач является процесс ETL. Данный процесс представляет собой комплекс операций, реализующих процесс переноса первичных данных из различных источников в аналитическое приложение или поддерживающее его хранилище данных. Является составной частью этапа консолидации данных в анализе данных. При этом, увеличение эффективности обработки не влияет на целостность и безопасность данных.

### IV. СПИСОК ЛИТЕРАТУРЫ

1. Использование MS SQL Server Analysis Services 2008 для построения хранилищ данных-2008 / В. В. Полубояров. – Национальный Открытый Университет "ИНТУИТ" –2016. – 663 с.
2. Microsoft SQL Server 2005 Analysis Services. OLAP и многомерный анализ данных / Э. Меломед, В. П. Степаненко / БХВ-Петербург. – 2007. – 586 с. 5–13.
3. Импорт данных из Excel в SQL Server или базу данных [Электронный ресурс] / Microsoft. – Режим доступа: <https://docs.microsoft.com/ru-ru/sql/relational-databases/import-export/import-data-from-excel-to-sql?view=sql-server-ver15>. – Дата доступа: 17.10.2021.
4. Что такое хранилище данных? [Электронный ресурс] / PureStorage. – Режим доступа: <https://www.purestorage.com/ru/knowledge/what-is-data-warehouse.html>. – Дата доступа: 17.10.2021.
5. ETL: что такое, зачем и для кого [Электронный ресурс] / ЧЕРНОБРОВОВ АЛЕКСЕЙ АНАЛИТИК. – Режим доступа: <https://chernobrovov.ru/articles/etl-cto-takoe-zachem-i-dlya-kogo.html>. – Дата доступа: 17.10.2021.

# ИЗВЛЕЧЕНИЕ ГРАФА СОЕДИНЕНИЙ КОМПОНЕНТ ИЗ ОПИСАНИЯ СБИС НА ЯЗЫКЕ VHDL

Черемисинова Л. Д., Черемисинов Д. И.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: {cld, cher}@newman.bas-net.by

*Рассматривается задача, возникающая при интеграции программ САПР от разных поставщиков в среду проектирования СБИС. При интеграции в среду проектирования существенным ограничением является недоступность внутренних функциональных возможностей программы, так что взаимодействие программ в процесс проектирования состоит только в обмене данными. Программы САПР взаимодействуют, обмениваясь сообщениями, представляющими собой описание проектируемого устройства на языке проектирования аппаратуры или в специальном обменном формате. Интерфейсом для объединения инструментов САПР может служить язык обмена информацией, в качестве которого часто используется язык описания аппаратуры VHDL. Предлагается метод и программа, которые извлекают граф соединенный компонент из описания СБИС на VHDL. Граф соединенный может быть использован для построения структурного описания в формате, требуемом сопрягаемой программой САПР.*

## ВВЕДЕНИЕ

Наиболее известными и широко используемыми языками описания оборудования (Hardware description language - HDL) систем автоматизации проектирования (САПР) являются языки Verilog и VHDL [1]. На этих языках можно описать как первоначальный проект СБИС, так и его промежуточные представления в процессе подготовки СБИС к изготовлению. Поскольку на HDL можно описывать как структуру, так и функции устройства, HDL часто используются для представлений промежуточных результатов проектирования, формируемых программами САПР СБИС. Для больших проектов неизбежно возникают проблемы, связанные с комплексированием инструментов САПР, которые решаются путем создания специальных программ сопряжения. В задачу этих программ входит анализ и изменение описания СБИС на HDL.

Анализ VHDL – это весьма сложная задача из-за синтаксической сложности этого языка [3]. Существуют несколько коммерческих анализаторов VHDL, которые позволяют исследователям, проектировщикам и другим пользователям строить инструменты проектирования и использовать VHDL-описания как обменный формат. Интерфейсы этих анализаторов отличаются по формату данных для представления результатов анализа и методам доступа к этим данным. Семантически текст на языке VHDL описывает некоторое цифровое устройство. В процессе проектирования используются несколько описаний, отличающихся уровнем абстракции поведения устройства. Фактически описания разного уровня абстракции отличаются языковой конструкцией, называемой в VHDL архитектурой. Каждый уровень требует особых языковых средств задания архитектуры, составляющих некоторое подмножество исходного языка описания проектирования аппаратуры.

## I. СТИЛИ КОДИРОВАНИЯ АРХИТЕКТУРЫ УСТРОЙСТВА НА VHDL

Языковые средства VHDL для задания архитектуры определенного уровня абстракции, составляют некоторое подмножество VHDL. Эти подмножества называются стилями кодирования. Основных стилей кодирования – три: 1) поток данных, 2) поведенческий. 3) структурный. Подмножества языка VHDL не составляют особых языков и заданы как рекомендации для использования проектировщиками с определенной целью и для фиксации результатов этапов проектирования. Для архитектуры структурного стиля характерно использование операторов создания экземпляров компонентов (port map). Стиль потока данных (data flow) характеризуется тем, что используются оператор назначения сигналов, логические и арифметические операторы. Стиль для описания объекта проектирования в алгоритмической форме путем задания преобразования входных данных в выходные, называется поведенческим. Элементы языка VHDL, которые могли бы быть синтезированы средствами синтеза (компиляторами) на уровне регистровых передач (register transfer level) составляют синтезируемое подмножество VHDL. В описании на синтезируемом подмножестве могут смешиваться все три стиля кодирования архитектуры.

## II. ЗАДАЧА СИНТАКСИЧЕСКОГО АНАЛИЗА СИНТЕЗИРУЕМОГО ПОДМНОЖЕСТВА VHDL

В теории формальных языков синтаксический анализ (разбор, парсинг – parsing) – это процесс определения того, может ли строка символов быть получена из множества продукций (set of production rules), называемых грамматикой. Однако, часто задачей синтаксического анализа заданного текста (последовательности символов) считается построение дерева вывода. Дерево вывода описывает синтаксическое стро-

ение текста в соответствии с правилами заданной грамматики. Для языка VHDL источником грамматики может быть стандарт языка – LRM (Language Reference Manual), задающий синтаксис и семантику языка. Граф соединений компонент может быть построен в результате обхода части дерева вывода, соответствующей архитектуре структурного (или смешанного) стиля.

Выводом (derivation) называется последовательность шагов, начинающаяся с начального символа грамматики, и заканчивающаяся строкой из терминальных символов. Каждый шаг состоит в применении правила грамматики, т.е. замене в исходной строке символа нетерминала правой частью правила. Строки, получаемые в результате применения правил грамматики, называются сентенциальными формами. Вид дерева вывода для заданных текста и грамматики зависит от метода синтаксического анализа – стратегии разбора. Имеются два класса стратегий: 1) левосторонняя (LL): для применения правила всегда выбирается самый левый нетерминал сентенциальной формы, 2) правосторонняя (LR): для применения правила всегда выбирается самый правый нетерминал сентенциальной формы. Синтаксический анализ языка VHDL все еще не является решенной проблемой на практике, несмотря на изощренность современных стратегий синтаксического анализа и долгую историю академических исследований.

Для детерминированности результата грамматики должна удовлетворять правилам избранной стратегии синтаксического анализа. Тип синтаксического анализатора со стратегией снизу вверх (LR парсер) распознает любую строку контекстно-свободного языка за линейное время. Для применения этой стратегии нужна леворекурсивная грамматика. Нисходящий анализ (LL парсер) – это стратегия определения принадлежности входной строки к некоторому формальному языку, в которой правила формальной грамматики применяются, начиная с начального символа грамматики, пытаюсь корректно разбить исходную строку на непосредственные составляющие. Для применения этой стратегии нужна праворекурсивная контекстно-свободная грамматика. Нисходящий рекурсивный парсер неприменим для разбора леворекурсивных грамматик. Праворекурсивную грамматику в нисходящей стратегии можно применить, если читать исходный текст в обратном порядке.

### III. АНАЛИЗАТОР VHDL, ГЕНЕРИРУЮЩИЙ ГРАФ СОЕДИНЕНИЙ КОМПОНЕНТ

Известны методы и программы, которые по заданной контекстно-свободной грамматике могут построить программу, генерирующую дерево вывода для заданного текста, удовлетворяющего грамматике. Эти программы требуют определенной формы грамматики. Автоматическое созда-

ние синтаксического анализатора, выполняющего анализ за линейное время, возможно только для подмножества контекстно свободных грамматик. Почти для любой произвольной грамматики можно найти немного другую грамматику, которая генерирует тот же язык и допускает синтаксический анализ за линейное время. Однако поиск такой грамматики почти всегда требует вмешательства человека и не может быть автоматизирован.

Генераторы анализаторов могут построить более компактную форму результатов анализа, вместо дерева синтаксического анализа для пост-синтаксической обработки, если модифицировать грамматику добавлением действий с побочным эффектом, создающих в памяти представление результатов в требуемой форме – в виде графа соединений компонент. Эта модификация грамматики требует вмешательства человека и тоже не может быть автоматизирована.

Наиболее известный генератор синтаксических анализаторов по заданной грамматике называется компилятором компиляторов yacc - bison. Этот компилятор компиляторов использует стратегию синтаксического анализа, называемую LALR(1). Аббревиатура означает синтаксический анализ «LR с просмотром на 1 символ вперед». Грамматика из LRM VHDL имеет форму, не подходящую для стратегии LALR(1) [2]. Так как интерфейсом для объединения инструментов САПР служит синтезируемое подмножество языка VHDL, можно исследовать грамматику для распознавания этого подмножества. Большинство приведенных в литературе грамматик леворекурсивны и требуют генераторов, использующих стратегию LL.

В результате опробования нескольких грамматик из литературы, подходящим для компилятора yacc - bison оказалась грамматика из открытой САПР СБИС Alliance [3]. Эта грамматика была дополнена действиями, позволяющими создать в памяти представление результатов анализа в виде графа соединений компонент.

## IV. ЗАКЛЮЧЕНИЕ

Разработанная программа анализа синтезируемого подмножества VHDL была использована в качестве компонента конвертера, преобразующего описание СБИС в форму на языках VHDL, EDIF, SF.

## V. СПИСОК ЛИТЕРАТУРЫ

1. VHDL'93. IEEE Standard VHDL Language Reference Manual. -IEEE Std 1076-1993, 2002. - 289 p.
2. Lorenc, Lubos et al. A Note on the Parsing of Complete VHDL-2002 // Workshop on Formal Models (2007) [Электронный ресурс] / <http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-255/paper11.pdf>
3. Alliance – A Free VLSI/CAD System [Электронный ресурс] / <http://www-soc.lip6.fr/equipement/logiciels/alliance>.

# ЗАДАЧА КЛАСТЕРИЗАЦИИ ПРИ РАСПОЗНАВАНИИ ЭНДОСКОПИЧЕСКИХ СНИМКОВ

Шилин Л. Ю., Козарь Р. В., Навроцкий А. А.

Кафедра информационных технологий автоматизированных систем,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь

E-mail: dekfitu@bsuir.by, pozitr0n.kozarroman@gmail.com, navrotsky@bsuir.by

*В данной работе представлены результаты анализа существующих на данный момент методов кластеризации данных медицинских снимков, полученных методом оптической эндоскопии. Также предложена модификация алгоритма Виолы-Джонса, учитывающая кластеризацию.*

## ВВЕДЕНИЕ

В настоящее время все также актуальны задачи по принятию решений, которые зависят от того, насколько объект, расположенный на медицинском изображении, будет подлежать классификации. К таким изображениям можно смело отнести эндоскопические снимки. Способность распознавания – это основное свойство всех биологических существ. Информационные же системы этим свойством не обладают в полной мере, поэтому основой алгоритма Виолы-Джонса для распознавания подобного рода объектов является выделение так называемых локальных признаков, присущие каждому изображению и, далее, последующего обучению на них алгоритма. Также для определения локальных признаков используются так называемые каскады Хара, которые очень эффективно работают в задачах распознавания [1]. Сохранение маленьких деталей изображения – исключительно важная составляющая при генерации изображений как в трехмерной графике, так и при распознавании изображений. Разумеется, эндоскопические снимки не являются исключением, поскольку обработка таких изображений выполняется при помощи метода скользящего окна  $U$  [2]. Поскольку искомый объект в его изначальном состоянии может иметь различный масштаб, необходимо выполнить поиск этого объекта с признаками различного масштаба. Это позволит выполнять однотипные вычисления на разных областях искомого изображения, а также на различных масштабах его признаков. Однако использование метода Виолы-Джонса в сочетании с методом скользящего окна имеет ряд недостатков. Главные из них: длительное время обучения при обработке и большое количество расположенных друг к другу результатов (причина в применении различных масштабов и метода скользящего окна). В качестве метода распознавания эндоскопических снимков используется метод распознавания Виолы-Джонса.

## I. ОПИСАНИЕ АЛГОРИТМА

Данный алгоритм анализирует каждую область изображения и находит однозначное ре-

шение о том, принадлежит ли искомый объект к рассматриваемой области. Те области, которые прошли через весь каскад, можно классифицировать только тогда, когда корректно будут классифицированы все их прецеденты [3]. Отклик такого признака  $f_j(x)$  можно вычислить как разность интенсивностей пикселей как в светлой, так и в темной областях. Важно отметить тот факт, что базовый алгоритм оперирует таким понятием, как слабый классификатор  $h_j(x)$ . Однако данный метод имеет один существенный недостаток: результат работы данного метода – огромное количество данных, которое обусловлено применением скользящего окна и спецификой самого эндоскопического снимка. Поэтому целью настоящей работы стала разработка более эффективного метода, который объединил бы большое количество разобренных данных, полученных на базе алгоритма Виолы-Джонса и анализа полученных результатов при помощи методов кластеризации. В оригинальном методе решение о том, содержится ли основной объект в рассматриваемом скользящем окне, принимается однозначное решение. Либо объект присутствует, либо нет. Данное решение принимается на основании формулы (2). Модификация заключается в другом «нечетком» подходе, суть которого в том, что результат будет «нечетким» и на его основе можно будет принять другое решение (решение является неоднозначным). В проведенных опытах было обнаружено, что при обработке медицинских изображений в некоторых условиях алгоритмом Виолы-Джонса принималось решение о том, что искомого объекта нет на текущем скользящем окне, в то время как в реальности объект существовал.

## II. МОДИФИЦИРОВАННЫЙ АЛГОРИТМ

Основная идея заключается в «неточной» оценке нахождения объекта в рассматриваемой области. Идея предложенного метода заключается в проверке и подтверждении следующей гипотезы: при уменьшении порога срабатывания алгоритма распознавания и применяя данный метод кластеризации объектов становится возможным улучшить эффективность его распознавания. Для сравнения традиционно исполь-



зуются статистические критерии, в частности,  $t$ -распределение (более известное, как распределение Стьюдента) [4]. Также в исследовании используются критерии оценки и будет предложен новый критерий на базе алгоритмов эталонных данных (Ground Truth) и элементов ROC-анализа (the Receiver Operator Characteristic). Оба этих метода представляют собой аппарат для анализа качества построенных моделей, а также активно используются для построения моделей в медицине и проведения клинических исследований. Также при этом следует отметить, что параметры в алгоритме кластеризации подбираются эмпирически. Также, проведя серию экспериментов был сформирован следующий график зависимости, по которому была точно определена оптимальная пара значений для алгоритма кластеризации при обработке данных медицинских изображений (а именно количество соседей, минимальное расстояние между соседями) и проиллюстрирована зависимость Ground Truth в зависимости от порога и параметра кластеризации (см. рис. 1).

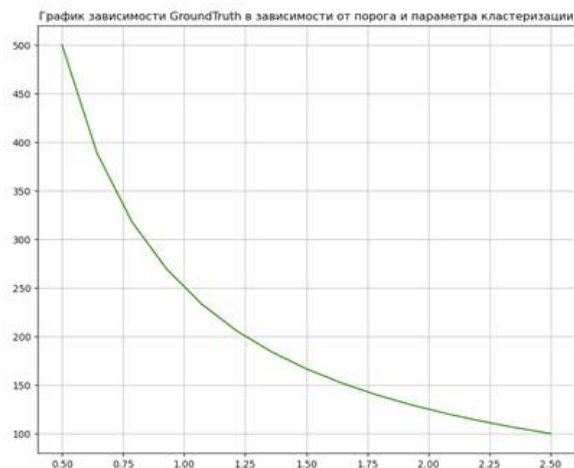


Рис. 1 – График зависимости Ground Truth в зависимости от порога и параметра кластеризации

### III. ОЦЕНКА АЛГОРИТМА. ВЫВОДЫ

Также, учитывая вероятность ложного срабатывания, представленного в табл. 1, необходимо выбрать подходящие параметры.

Таблица 1 – Вероятность ложного срабатывания в зависимости от порога

| Порог распозн.      | 0,80 | 0,85 | 0,89 | 0,95 | 0,99 | 1 |
|---------------------|------|------|------|------|------|---|
| Ложн.срабыв (проц.) | 98   | 18   | 14   | 5    | 1    | 0 |

В данной статье исследована эффективность использования алгоритма Виолы-Джонса для распознавания медицинских изображений. Определено, что при использовании этого метода при распознавании и обработке медицинских изображений необходимо большое количество выходных данных алгоритма распознавания и длительное время работы, требуемое для обучения. Установлено то, что использование нового критерия оценки дает однозначный ответ о принадлежности рассматриваемой области к искомому объекту. Полученные результаты убедительно показали, что предложенная модификация алгоритма на 58 процентов эффективнее базового алгоритма Виолы-Джонса справляется с ошибками 1 рода и на 77 процентов эффективнее справляется с ошибками 2 рода (см. рис. 2).

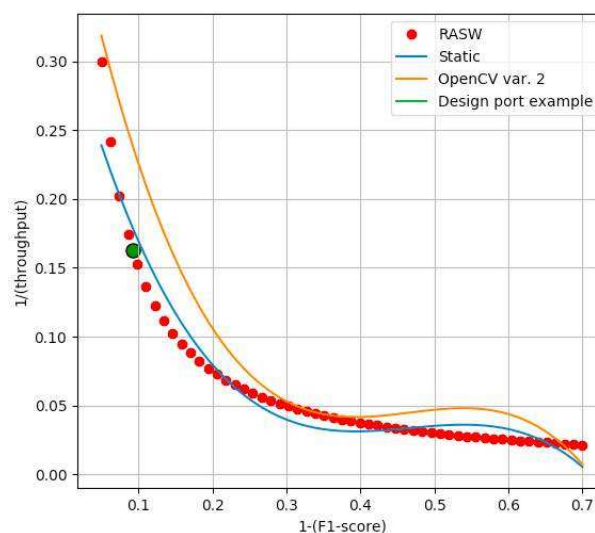


Рис. 2 – Оценка эффективности работы метода кластеризации

### IV. СПИСОК ЛИТЕРАТУРЫ

1. Viola P., Jones M. J. Robust real time face detection // International Journal of Computer Vision. – 2004. – V. 57. – № 2. – P. 137–154.
2. Гонсалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2005. – 1072 с.
3. Местецкий Л. М. Математические методы распознавания образов. – М.: МГУ, ВМиК, 2002–2004. – С. 20–24.
4. Шапиро Л. Компьютерное зрение / Л. Шапиро, Д. Стокман. – М.: Изд. Дом «Лаборатория знаний», 2015. – 763 с.

# MODEL-BASED OPTIMIZATION OF THE REGRESSION OBJECTS

Mukha V. S., Kako N. F.

Department of Information Technologies of Automated Systems,  
Belarusian State University of Informatics and Radioelectronics

Minsk, Republic of Belarus

E-mail: mukha@bsuir.by, kako.nancy@gmail.com

*The model-based procedure for determining of the extremum of the regression object is proposed. The numerical comparative analysis of this procedure and Kiefer–Wolfowitz stochastic approximation procedure is performed.*

## INTRODUCTION

Stochastic approximation and random search are most known optimization method at present for extreme objects. They are so called search methods. These approaches do not use the mathematical model of the object. In opposite to the search methods, the model-based approach suppose designing the mathematical model of the controlled object. This allows us to obtain then the mathematical model of the optimality criterion and to find the optimal control action.

### I. STOCHASTIC APPROXIMATION

The stochastic approximation is a method for solving a wide range of estimation problems based on recurrent refinement of the estimate with an increase in the number of observations [1, 2, 3]. The first procedure of stochastic approximation was Robbins–Monro stochastic approximation procedure, designed to find the zero of an unknown regression function. The Kiefer–Wolfowitz procedure of stochastic approximation was designed to find the extremum point of an unknown regression function. The stochastic approximation procedures are based on the sequentially estimating, from one observation to next. We consider the Kiefer–Wolfowitz procedure more detail.

Let  $\phi$  be measurable scalar function on  $R^n$ . We suppose, that we can measure the realization of the random variable

$$y(X) = \phi(X) + \epsilon(X)$$

in any point  $X \in R^n$ , where the random errors  $\epsilon(X)$  are centered ( $E(\epsilon(X)) = 0$ ) and mutually independent for any  $X_0, X_1, \dots \in R^n$ , is the regression function. The problem consist of the finding the sequence of the points  $X_0, X_1, \dots \in R^n$  converging to the point  $X^*$ :

$$X^* = \arg \max_{X \in R^n} \phi(X).$$

The symmetric Kiefer–Wolfowitz algorithm has the following form [2]:

$$X_{k+1} = X_k + \gamma_{k+1} \alpha_{k+1}^{-1} \times$$

$$\times \sum_{i=1}^n (y(X_k + \alpha_{k+1} e_i) - y(X_k - \alpha_{k+1} e_i)) e_i, \quad (1)$$

where  $k = 0, 1, 2, \dots, K$  and  $e_i$  are orts, i.e. coordinate vectors  $e_i = (0_1, \dots, 1_i, \dots, 0_n)^T \in R^n$ .

The sequences  $\gamma_k = 1/k$ ,  $\alpha_k = 1/\sqrt[3]{k}$  provide the convergence of the Kiefer–Wolfowitz algorithm.

## II. RANDOM SEARCH

The random search is an optimization method that was proposed for finding of the extremum point of an unknown deterministic inertialess object (function). The term “random search” is attributed to Rastrigin [4]. The distinctive feature of this method is that the points are selected in the search space as random numbers from a certain probability distribution. It is considered that the random search is the most effective method for finding the global extremum. The simplest random search (so named blind scan) for minimization is as follows. The values of the function are calculated in random points and the minimum value is chosen.

The random search can be used in regression experiment for the mathematical model object building. However, the data processing is performed in such an application in package form, i.e. not in the real observation time. That is the irresistible disadvantage of the random search.

## III. MODEL-BASED METHOD

The model-based approach suppose designing the mathematical model of the controlled object. This allows then to obtain the mathematical model of the optimality criterion and to find the optimal control action. This approach is put into dual control theory [5, 6, 7]. The Box–Wilson method in the framework of the extreme experiment design [8, 9] includes the model-based approach too, but it is rather methodology than a method or algorithm, so it is difficult to implement.

It is most simple to approximate the controlled object by quadratic polynomial.

Let  $X = (x_{j(q)})$ ,  $j(q) = (j_1, j_2, \dots, j_q)$ , be a  $q$ -dimensional matrix, that is the argument of a scalar function  $\phi(X)$ , and this function has the form

$$y = \phi(X) = C_0 + {}^{0,q}(C_1 X) + {}^{0,2q}(C_2 X^2), \quad (2)$$

where  $C_k$ ,  $k = 0, 1, 2, \dots$ , are the  $kq$ -dimensional-matrix parameters of the function  $\phi(X)$  and  $C_2$  is symmetric relative its last  $q$ -multi-indexes [10]. Let it be required to find the extremum of this function.

Optimal value of  $X$  can be found from the equation  $d\phi(X)/dX = 0$ . Differentiating of (2) gives the equation  $C_1 + 2^{0,q}(C_2X) = 0$ , from which we get

$$X^* = -^{0,q}(C_2^{-1}C_1)/2, \quad (3)$$

where  $^{0,q}C_2^{-1}$  is the matrix  $(0, q)$ -inverse to the matrix  $C_2$ .

Let us to find the minimum value  $y^* = \phi(X^*)$  of the function  $\phi(X)$ . Because of the fact that  $^{0,2q}(C_2X^2) = ^{0,q}(C_2X)X$  and  $^{0,q}(C_2X^*) = -C_1/2$ , we have  $^{0,2q}(C_2(X^*)^2) = -^{0,q}(C_1X^*)/2$  and

$$y^* = \phi(X^*) = C_0 + ^{0,q}(C_1X^*). \quad (4)$$

Substituting  $X^*$  (3) into (4), we obtain

$$y^* = C_0 - ^{0,q}(C_1 ^{0,q}(C_2^{-1}C_1))/4.$$

We can now perform  $K$  experiments in  $K$  given points  $X_1, X_2, \dots, X_K \in R^n$  and find the estimations  $\hat{C}_0, \hat{C}_1, \hat{C}_2$  of the parameters  $C_0, C_1, C_2$ . Then we get the following point of the extremum:

$$X^* = -\frac{1}{2} ^{0,q}(C_2^{-1}\hat{C}_1).$$

The points of experiments  $X_1, X_2, \dots, X_K$  may be regular or random as in the random search approach.

#### IV. COMPUTER SIMULATION

We have performed computer check of the symmetric Kiefer–Wolfowitz algorithm. We simulated the regression object with the regression function of the form

$$\phi(X) = 1 + x_1 + 2x_2 + 2x_1^2 + 4x_2^2$$

and with normal distributed inner noise  $\epsilon(X)$  with zero mean value and variance equal to 0.2. The parameters of this regression function in multidimensional-matrix form (2) are as follows:

$$C_0 = 1, \quad C_1 = \begin{pmatrix} 1 & \\ & 2 \end{pmatrix}, \quad C_2 = \begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix}.$$

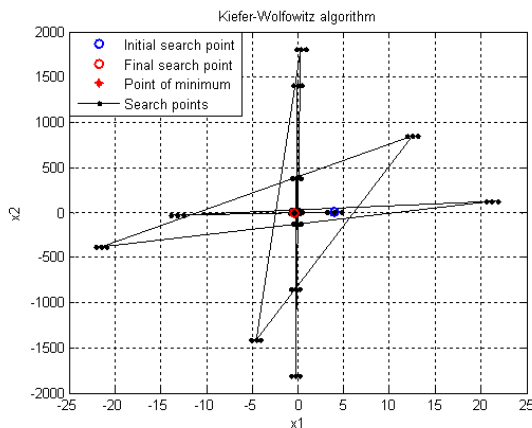


Fig. 1 – The points of experiments in the symmetric Kiefer–Wolfowitz algorithm

Figure 1 shows the search process when iterations number equal to 25 ( $K = 25$  in (1)). The real minimum point of the regression function equal to  $(-0.25, -0.25)$ , the found point equal to  $(-0.03, -0.26)$ . We can see that Kiefer–Wolfowitz algorithm searches the minimum point quite well. However, the Kiefer–Wolfowitz algorithm requires very big number of experiments (125 provided  $K = 25$ ). This makes serious difficulties for utilization it in a real production control process, since each experiment is accompanied by a product release. It can be very expensive.

Figure 2 shows the points of measurements in model-based approach with the number of the measurements equal to 25. The measurements were processed by classical least square method. The accuracy is very high.

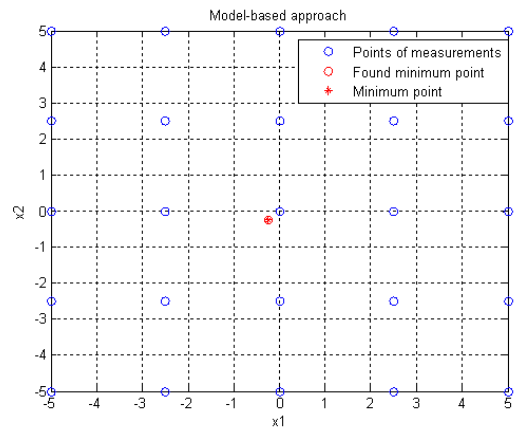


Fig. 2 – The points of experiments in the model-based approach

#### V. REFERENCES

1. Probability and mathematical statistics: Encyclopedia. Editor-in-chief Yu.V. Prokhorov. Moscow, Big Russian Encyclopedia, 1999, 910 p. In Russian.
2. Ermakov S.M., Zhiglyavskii A.A. Mathematical Theory of the Optimal Design. Moscow, State Press for Physico-Mathematical Literature, 1987, 318 p. In Russian.
3. Wasan M.T. Stochastic Approximation. Cambridge university press, 2004, 216 p.
4. Rastrigin L.A. Systems of extreme control. Moscow, Nauka, 1974. 630 p. In Russian.
5. Feldbaum A.A. Optimal Control Systems. Academic Press, New York and London, 1965. 452 p. Mathematics in Science and Engineering. A series of monographs and textbooks. Edited by Richard Bellman. Volume 22.
6. Mukha V.S. On the dual control of the inertialess objects. Proceedings of the LETI. Issue 130, 1973. Pp. 31–37. In Russian.
7. Mukha V.S., Sergeev E.V. Dual control of the regression objects. Proceedings of the LETI. Issue 202, 1976. Pp. 58–64. In Russian.
8. Box G.E.P., Wilson K.B. On the Experimental Attainment of Optimum Conditions. Journal of the Royal Statistical Society. 1951. Vol. 13, No. 1. Pp. 1–45.
9. Box G.E.P., Hunter W.G. and Hunter J.S. Statistics for experimenters. New York: Wiley, 1978. 653 p.
10. Mukha V.S. Analysis of multidimensional data. Minsk, Technoprint, 2004. 368 p. In Russian.

# SELECTING ANSWERS FROM CV TEXT

Nasr S.

Information technologies in automatized systems department,  
The Belarusian State university of informatics and radioelectronics  
Minsk, Republic of belarus  
E-mail: sara.nasr@gmail.com

*The problem of extracting answer(s) from text block is considered. The text represents a Curriculum Vitae (CV) divided into semantic blocks. The problem is to find in CV the corresponding block containing applicant data to be extracted and inserted in some predefined form (PF) for automatic processing. So, the required semantic block should be found and processed appropriately to deliver necessary information for PF. The method is outlined to solve this problem.*

## INTRODUCTION

One of the interesting applied problems is automatic CVs processing. The final goal is to select the best candidate for some position(s) from those who sent his/her e-mail. Typically, CV contains a block of private and professional data of the candidate including age, education, participating in real projects, availability of published papers or research activities, possessing modern programming languages and technologies and so on. The problem of selection of the best candidate requires to estimate the integral choice function usually represented in the next form [1]

$$CF = \sum w_i \cdot uf_i.$$

Here,  $w_i$  stands for the weight of criterion  $i$ , and  $uf_i$  represents the value of utility function of criterion  $i$ . One of the basic problems is connected to extracting data from CV to be used in evaluation of the functions  $uf_i$ . We shall restrict our consideration by the demonstration of a common approach to solving the above problem. Namely, we suppose that CV is previously divided into semantic blocks in a way that to get answer to some question like «what is your age?» or «when you were born?» is possible by means of some text processing procedure outlined later on in this report. Of course, to divide text into semantic blocks is some serious applied problem as well. One of its possible solution is based on computing the correlation co-efficients between keywords in CV text and define semantic blocks as those, containing the subset of keywords mostly correlated to each other. We, however, leave this question without detail explanation. So, the main idea of the report consists in preparing some list of questions to CV text needed for filling PF. Clearly, the specificity of this process is its extreme undefiniteness. The rest of this short report contains the details.

## I. DEMONSTRATION OF THE APPROACH

As was said earlier, we require to prepare a list of questions to get the desired information. As we consider here an example with applicant age

definition, let the corresponding list to consists of the next questions:

- Q1. What is your age?
- Q2. How old are you?
- Q3. What is your date of birth?
- Q4. When were you born?
- Q5. What is the date of your borning?
- Q6. How much years do you have?

The question is asked if and only if the previous question failed to be answered. This means that CV text does not contain the text block identified as suitable for (answering) the question. In order to be more concrete, let us consider the following CV text block in the field of our interests: «I am 25 years old and was born 1n 1996». This block contains even extra information for our needs. Now (leaving explanation for next section) the first question Q1 remains unanswered. But Q2, and Q4 with Q6 are success. It is enough to find data delivered by Q2 what prevents asking Q4. Now let us explain the idea of the method.

## II. METHOD DEFINITION

We have to explain the method of CV-text processing. The entire process is divided into four stages in general [2]. The first stage is to transform Word document or a pdf-file with CV to a plain text. This can be done with a Tika system which extracts a raw text and deletes unnecessary control information such as colors, fonts, and the like. The next step is to get keywords of the text. The idea is to consider practically all text words as keywords due to not big size of CV. It is required that each sentence in CV contains at minimum one keyword. Besides, this simplifies the algorithm as it not requires to define each keyword score and test that each sentence is covered at least by one keyword. The prepositions, conjunctions, pronouns, auxiliary and modal verbs (such as can, have, may etc.) are excluded. For an example above, the set of keywords contains the words years, old, born. The keywords are selected in such a way that the similar words are identified as the same. For instance, programming and programmer are considered as one keyword. By thus, each keyword labels one or more semantic text blocks. To realize the keywords

selection we use the Dice metrics (measure) given by the formula

$$q = \frac{2|X \cap Y|}{|X| + |Y|}.$$

This formula estimates similarity of two sets X and Y,  $|X|$  denotes the number of elements in X;  $X \cap Y$  denotes two sets intersection. From practical viewpoint, the value of  $q$  should be not less than 0.5 (however, the experiments are required since this co-efficient depends of the compared sets sizes quite essentially). In the above formula, X stands for the set of keywords of some semantic block, and Y represents a question keywords. Again, pay attention to the fact that CV is short text, so practically each word of CV may be regarded a keyword.

The next stage consists of building two directories. Each directory represents a collection of the pairs  $\langle \text{key}, \text{value} \rangle$ . The first directory contains the pairs of items and each pair represents a record with key standing for a keyword and a value representing the set of sentences (text blocks) numbers labelled with the keyword key. The second dictionary consists of the pairs representing the sentences numbers and their texts. Now let us explain how to extract an answer to the question represented by a set of words (some of words are keywords and some – not (these latter words do not belong to CV text)). For each keyword  $k_i$  in the question the set of sentences numbers  $N_i = n_{i1}, n_{i2}, \dots, n_{iz}$  is defined. Then for all  $N_i$  the most frequently encountered number(s)  $n_w$  is (are) defined. This number  $n_w$  defines a sentence (block) to be considered as an answer. If there are more than one candidate to be an answer then all candidates are considered. In the example the only keyword «old» is common in question and text block in resume («I am 25 years old and was born 1n 1996»). This gives Dice estimation equals to 0.67. By this, the text block  $B$  with the required data is defined.

The next step consists in processing  $B$  to extract the required data. Of course, the question requires some commonly adopted form of the answer as, for example, «I am 25 years old» or «I am twenty five years old». The main feature of these two sentences is availability of the word «old». To extract age of the applicant from the text block defined, the grammar parsing technique is used, associated with the text of the answer. The text of the question («How old are you») supposes the following possible answer structures:

- A1. I am ? years.
- A2. I am ? years old.

- A3. I am ?.
- A4. Me is ? years.
- A5. Me is ? years old.
- A6. Me is ?.
- A7. am ? years.
- A8. am ? years old
- etc.

Here, «?» stands for the data we are looking for. So, each answer pattern should be checked. Again, the Dice metrics is applied to evaluate degree of closeness of the resume text of block textitB and Ai. To identify the the numeric value of the applicant age is a simple string processing task.

### III. CONCLUSION

Let us summarize the general approach to extracting data from CV,

1. The sets of possible questions to define each data item in resume should be prepared in advance. This means, that some set of questions should relate to age, some to education, some to professional skills *etc.*
2. Each question is associated with some set of keywords (the keywords are directly defined from the question).
3. The system finds the textblock (semantic block) in CV which is close to the question with respect to Dice metrics (obviously, there are other metrics, besides Dice metrics, as well. However, selection of the metrics is not crucial for the system work).
4. After a semantic block is found, the data should be extracted. This is done by means of the previously created number of answer patterns. Each pattern should be considered with respect to its closeness to the semantic block. By means of the Dice metrics we are in position to define the required pattern and extract data accordingly to this pattern.

The technique outlined in this report is used as an essential part of the CV-processing system. It admits availability of the text mistakes and inaccuracies as well.

One of the important issues concerns definition of the optimal set of features to be analyzed in CV. This question is also of our research interests.

### IV. REFERENCES

1. Saati, T.L. Decision making with the analytic hierarchy process / T. L. Saati // Int. J. Services Sciences, Vol. 1, No. 1, 2008 .
2. German, Yu. Information extraction method from resume / Yu. O. German, O. V. German, S. Nasr // Proceedings of BSTU Scientific journal. (Minsk, Belarus). 2019, №1(218), p.p.64–69.

# Содержание

|                                                                                                                                                                                           |           |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b>Пленарные доклады</b> . . . . .                                                                                                                                                        | <b>7</b>  |
| <b>Авдеев Н. А., Глебов С. К., Заливако С. С., Иванюк А. А., Пинчуков Д. А., Семенов В. В., Цибулько Ю. В.</b><br>Проектирование и верификация контроллеров SSD-дисков . . . . .          | <b>8</b>  |
| <b>Иванюк Д. С., Пупена А. Н., Шункевич Д. В.</b><br>Онтологический подход к комплексной автоматизации предприятий рецептурного производства в контексте концепции Industry 4.0 . . . . . | <b>17</b> |

|                                                                                                                                                                                                  |           |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b>Секция «Системы управления»</b> . . . . .                                                                                                                                                     | <b>27</b> |
| <b>Боровиков С. М., Казючиц В. О., Дик С. С., Юреня К. В.</b><br>Модель прогнозирования времени тестирования прикладных компьютерных программ для автоматизированных систем управления . . . . . | <b>28</b> |
| <b>Гринюк Д. А., Олиферович Н. М., Сухорукова И. Г., Оробей И. О.</b><br>Настройка ПИД-регулятора для системы с нелинейной динамикой . . . . .                                                   | <b>30</b> |
| <b>Довнар А. Д.</b><br>Постановка задач группового управления промышленными мобильными роботами . . . . .                                                                                        | <b>32</b> |
| <b>Довнар А. Д., Рогач А. Ю.</b><br>Кинематика и динамика ножничного подъемника . . . . .                                                                                                        | <b>34</b> |
| <b>Дорох А. А., Деменковец Д. В.</b><br>Мобильное программное средство для автоматизации взвешиваний железнодорожных вагонов . . . . .                                                           | <b>36</b> |
| <b>Кисель Т. В.</b><br>Особенности организации приемной кампании в вузах ряда стран СНГ . . . . .                                                                                                | <b>38</b> |
| <b>Клютко М. В., Кобринец В. П., Карпович Д. С.</b><br>Создание динамической модели электромеханической системы с использованием MATLAB и AUTOCAD . . . . .                                      | <b>40</b> |
| <b>Крищенко В.А., Захарьев В.А.</b><br>Системы верификации субъектов доступа на основе речевого сигнала . . . . .                                                                                | <b>42</b> |
| <b>Кузнецов А. П., Снисаренко С. В.</b><br>Обработка видеопотока в системе технического зрения . . . . .                                                                                         | <b>44</b> |
| <b>Курулёв А. П.</b><br>Оценка надежности фазосдвигающего генератора . . . . .                                                                                                                   | <b>47</b> |
| <b>Петровский И. И., Свито И. Л., Шилин Л. Ю.</b><br>Элементы высшего порядка в полосовых фильтрах . . . . .                                                                                     | <b>49</b> |
| <b>Слинка К. И.</b><br>Smart АЗС: понятие, состояние, перспективы . . . . .                                                                                                                      | <b>51</b> |
| <b>Хаджинов М. К., Павлова А. В.</b><br>Оценивание и компенсация параметрических возмущений сервопривода . . . . .                                                                               | <b>53</b> |

|                                                                                                                                                                                        |           |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b>Черемисинова Л. Д., Черемисинов Д. И.</b><br>Тестовая система для функциональной верификации дискретных устройств с параллелизмом поведения . . . . .                               | <b>55</b> |
| <b>Шаблинская Е. А., Деменковец Д. В.</b><br>Программно-аппаратное средство «Школьный журнал» на базе микрокомпьютера Orange Pi One с использованием бесконтактных RFID-карт . . . . . | <b>57</b> |
| <b>Шведова О. А.</b><br>Особенности реализации задач цифровизации на белорусских предприятиях . . . . .                                                                                | <b>59</b> |
| <b>Шумский А. Н., Карпович Д. С.</b><br>Система управления поддержания траектории полета бпла при мониторинге окружающей среды с использованием теории нечётких множеств . . . . .     | <b>61</b> |
| <b>Elzein Imad</b><br>Analysis of artificial neural network in a photovoltaic system to extract the maximum power in a photovoltaic based system paradigm . . . . .                    | <b>63</b> |



|                                                                                                                                                                          |           |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b>Секция «Интеллектуальные информационные технологии»</b>                                                                                                               | <b>67</b> |
| <b>Адамюк А. А., Щегрикович Д. В.</b><br>Разработка алгоритма добавления источников света в компьютерные сцены . . . . .                                                 | <b>68</b> |
| <b>Владова А. Ю.</b><br>Подходы к идентификации технического состояния верхнего строения железнодорожного пути . . . . .                                                 | <b>70</b> |
| <b>Воронова В. В., Удовин И. А.</b><br>Обзор методов генерации текста на естественном языке . . . . .                                                                    | <b>72</b> |
| <b>Врублевский С. С., Бысов А.А.</b><br>Разработка имитационной модели виртуальной частной сети электросвязи специального назначения в сетевом симуляторе ns-3 . . . . . | <b>74</b> |
| <b>Дворниченко А. А.</b><br>Методы анализа финансового рынка с применением нейросетевого подхода . . . . .                                                               | <b>76</b> |
| <b>Денисевич Д. А., Якимов Е. А.</b><br>Генетический алгоритм для раскрытия листовых материалов . . . . .                                                                | <b>78</b> |
| <b>Загорский А. Г.</b><br>Онтологический подход к интеграции сторонних функциональных приложений в системы обработки знаний . . . . .                                    | <b>80</b> |
| <b>Ивашенко В. П.</b><br>Моделирование решения задач прикладных интеллектуальных систем на интеграционной платформе . . . . .                                            | <b>82</b> |
| <b>Ивашенко В. П.</b><br>Технология разработки программных компонентов интеллектуальных систем на основе интеграционной платформы . . . . .                              | <b>84</b> |
| <b>Ивашенко В. П., Кальчевский Д. А.</b><br>Применение подхода к построению плана поиска в базе знаний на основе стоимости операций . . . . .                            | <b>86</b> |
| <b>Карпук А. А., Говорко А. В.</b><br>Построение прикладной онтологии сетей радиосвязи . . . . .                                                                         | <b>88</b> |
| <b>Лось П. В.</b><br>Принципы представления семантических моделей в графовых базах данных . . . . .                                                                      | <b>90</b> |
| <b>Новицкий В. А.</b><br>Модели, методы и средства построения формальных семантических моделей пользовательских интерфейсов . . . . .                                    | <b>92</b> |

|                                                                                                                                                            |            |
|------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Пилипчук Л. А., Малявко А. В.</b><br>Эффективные алгоритмы декомпозиции линейных недоопределенных систем с матрицей инцидентности мультиграфа . . . . . | <b>94</b>  |
| <b>Сергей П. Д.</b><br>Статистические методы анализа данных для обнаружения мошенничества . . . . .                                                        | <b>96</b>  |
| <b>Скобцов В. Ю.</b><br>Нейросетевые модели для бинарной классификации данных телеметрической информации малых космических аппаратов . . . . .             | <b>98</b>  |
| <b>Удовин И. А., Воронова В. В.</b><br>Применение алгоритмов для определения плагиата в программном коде . . . . .                                         | <b>101</b> |
| <b>Халецкий А. М.</b><br>Система оценки знаний учащихся с прогнозированием успеваемости . . . . .                                                          | <b>103</b> |
| <b>Li Wenzu, Qian Longwei.</b><br>An approach to calculating the similarity between semantic segments in the intelligent tutoring systems . . . . .        | <b>105</b> |
| <b>Qian Longwei, Li Wenzu</b><br>Implementation principles of knowledge acquisition for intelligent system . . . . .                                       | <b>107</b> |
| <b>Savenko A. G., Sherstnev A. S.</b><br>Model and algorithm for adaptive search by logical expressions . . . . .                                          | <b>109</b> |
| <b>Zhirko M., Titko V., Grakova N.</b><br>Segmentation of brain tumor multi-parametric MRI scans using artificial neural networks . . .                    | <b>111</b> |

|                                                                                                                                                                                                                        |            |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Секция «Проектирование встраиваемых систем» . . . . .</b>                                                                                                                                                           | <b>113</b> |
| <b>Азаров И.С., Качинский М.В., Петровский Н.А.,<br/>Рыбенков Е.В., Станкевич А.В.,<br/>Устройство для мониторинга радиочастотного спектра на базе Xilinx Zynq . . . . .</b>                                           | <b>114</b> |
| <b>Бибило П. Н., Кириенко Н. А.<br/>Схемная реализация модулярных сумматоров в заказных КМОП СБИС . . . . .</b>                                                                                                        | <b>116</b> |
| <b>Деменковец Д. В., Леванцевич В. А.<br/>Неразрушающее тестирование запоминающих устройств на базе двойных адресных последовательностей . . . . .</b>                                                                 | <b>118</b> |
| <b>Заливако С. С., Иванюк А. А.<br/>Методика сравнения генераторов случайных числовых последовательностей по результатам тестов NIST . . . . .</b>                                                                     | <b>120</b> |
| <b>Иванюк А. А., Заливако С.С.<br/>Реализация физически неклонированной функции на LUT-блоках FPGA . . . . .</b>                                                                                                       | <b>122</b> |
| <b>Кардаш С. Н.<br/>О функциональных разложениях связанных систем булевых функций . . . . .</b>                                                                                                                        | <b>124</b> |
| <b>Кохновский С. И., Иванюк А. А.<br/>Влияние длительности работы кольцевого осциллятора на статистические характеристики последовательности бит, сгенерированной аппаратным генератором случайных чисел . . . . .</b> | <b>126</b> |
| <b>Петровская В. В., Деменковец Д. В.<br/>Анализ тестов запоминающих устройств для обнаружения пассивных кодочувствительных неисправностей . . . . .</b>                                                               | <b>128</b> |
| <b>Шамына А. Ю., Иванюк А. А.<br/>Анализ характеристик физически неклонированных функций типа арбитр различных конфигураций . . . . .</b>                                                                              | <b>130</b> |
| <b>Шинкевич Н.Н., Шамына А.Ю.<br/>Анализ устойчивости физически неклонированной функции типа арбитр к криптографическим атакам с использованием нейронных сетей долгой краткосрочной памяти . . . . .</b>              | <b>132</b> |

|                                                                                                                                                                                |            |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Секция «Математическое моделирование и компьютерная графика»</b> . . . . .                                                                                                  | <b>135</b> |
| <b>Бондарева Т. О., Лапицкая Н. В.</b><br>Алгоритм выявления неисправностей механизма, основанный на преобразовании вибрационного сигнала . . . . .                            | <b>136</b> |
| <b>Герасимов В. А.</b><br>Разработка программного обеспечения автоматизированного рабочего места военнослужащего курсового звена . . . . .                                     | <b>138</b> |
| <b>Гуринович А. Б., Митьковец Л. В., Сидоров Д.</b><br>Модифицированный алгоритм сжатия данных . . . . .                                                                       | <b>140</b> |
| <b>Езовит А. В., Гуринович А. Б.</b><br>Методы выделения информационных признаков речевых сигналов . . . . .                                                                   | <b>142</b> |
| <b>Журавлёв В.И., Наумович Н.М., Бричкалевич П.И.</b><br>Параметрическая схема моделирования характеристик виброизоляторов для синтезаторов частот . . . . .                   | <b>144</b> |
| <b>Коршикова Д. В., Гуревич О. В., Кукин Д. П., Шатилова О. О.</b><br>Интерпретация некоторых аспектов теории вероятности при разработке компьютерных игр . . . . .            | <b>146</b> |
| <b>Красковский П. Н., Серебряная Л. В.</b><br>Метод отложенного затенения трёхмерных сцен, использующий буфер видимости . . . . .                                              | <b>148</b> |
| <b>Никульшина К. Б., Гуринович А. Б.</b><br>Нейросетевые алгоритмы кадрового планирования . . . . .                                                                            | <b>150</b> |
| <b>Носкова Е. Е., Дружинская Е. В.</b><br>Генетический алгоритм. Программная реализация этапа мутации . . . . .                                                                | <b>152</b> |
| <b>Осипов А. Н., Хазановский И. О., Котов Д. В., Фролов А. В., Пацеев А. В., Пацеев С. В.</b><br>Смарт-система персонального мониторинга состояния здоровья человека . . . . . | <b>154</b> |
| <b>Пашук А. В., Гуринович А. Б., Волорова Н. А.</b><br>Проектирование архитектуры распределенной системы удаленной печати . . . . .                                            | <b>156</b> |
| <b>Пилипчук Л. А., Романчук М. П.</b><br>О проблеме оценки однородного потока в двунаправленной сети . . . . .                                                                 | <b>158</b> |
| <b>Полячок Е. Н.</b><br>Численное исследование эффективности методов решения задач о кратчайших путях в ориентированных графах . . . . .                                       | <b>160</b> |

|                                                                                                                                                                                                                               |            |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Прорашнев В. Л.</b><br>Реализация алгоритмов триангуляции для твердотельного моделирования . . . . .                                                                                                                       | <b>162</b> |
| <b>Таранчук В. Б.</b><br>Методы и средства системы ГеоБазаДанных для адаптации компьютерных моделей. Инструменты кластеризации . . . . .                                                                                      | <b>164</b> |
| <b>Цегельник В. В.</b><br>Об одном свойстве решений семейства трехмерных пятиэлементных диссипативных динамических систем с хаотическим поведением . . . . .                                                                  | <b>166</b> |
| <b>Чигирь И. В.</b><br>Алгоритм работы углового дискриминатора РЛС точного измерения координат при воздействии КИМ-помехи . . . . .                                                                                           | <b>168</b> |
| <b>Шатилова О. О., Кукин Д. П., Гуревич О. В., Коршикова Д. В.</b><br>Прототипирование пользовательских интерфейсов . . . . .                                                                                                 | <b>170</b> |
| <b>Ширшова М.А., Дружинская Е.В., Фатхиев Т.И., Хасанов Р.Р.</b><br>Разработка комплекса виртуальных лабораторных работ (КВЛР) «Технологии сооружения переходов и сложных участков магистральных газонефтепроводов» . . . . . | <b>172</b> |
| <b>Ярмолик С. Н., Зайко Е. В., Свинарский М. В., Леонович А. С.</b><br>Последовательный алгоритм сигнально-траекторного радиолокационного распознавания с адаптацией к углам пространственной ориентации объекта . . . . .    | <b>174</b> |
| <b>Hussein A. T., Gurinovich A. B.</b><br>Enterprise resource planning for startups in Iraq . . . . .                                                                                                                         | <b>176</b> |
| <b>Williams E., Gurinovich A. B.</b><br>Nigerian banks data analytics . . . . .                                                                                                                                               | <b>178</b> |

|                                                                                                                                                               |            |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Секция «Автоматизированные системы обработки информации»</b> . . . . .                                                                                     | <b>181</b> |
| <b>Архипенко Я. С., Пилипенко В.В.</b><br>Система маркировки и прослеживаемости товаров, как метод борьбы с контрафактной продукцией . . . . .                | <b>182</b> |
| <b>Белощедов Е. С., Гуринович А. Б.</b><br>Применение машинного обучения для увеличения производительности СУБД PostgreSQL . . . . .                          | <b>184</b> |
| <b>Бранцевич П. Ю., Лапицкая Н. В., Леванцевич В. А.</b><br>Моделирование вибрационных сигналов с заданными характеристиками . . . . .                        | <b>186</b> |
| <b>Герман Ю. О.</b><br>Об одной модели интернет-бизнеса . . . . .                                                                                             | <b>188</b> |
| <b>Гоглев И. В.</b><br>Применение метода векторизации для анализа русскоязычной текстовой информации . . . . .                                                | <b>190</b> |
| <b>Гудков А. С., Навроцкий А. А.</b><br>Шахматный движок . . . . .                                                                                            | <b>192</b> |
| <b>Гринюк Д. А., Олиферович Н. М., Сухорукова И. Г., Карпович М. Д., Михновец А. С.</b><br>Использование нелинейного сглаживания для анализа данных . . . . . | <b>194</b> |
| <b>Денисик П. Д.</b><br>Улучшение надежности программных средств с помощью применения автоматизированного тестирования . . . . .                              | <b>196</b> |
| <b>Дмитриев А. С.</b><br>Алгоритм описания искусственного интеллекта для независимого игрового мира . . . . .                                                 | <b>198</b> |
| <b>Кобяк И.П.</b><br>Точное значение вероятности пропуска ошибки при наблюдении векторов переходов в асимптотике . . . . .                                    | <b>200</b> |
| <b>Колесников В. Г.</b><br>Модернизация алгоритма поставки обновлений для систем мейнфрейма . . . . .                                                         | <b>202</b> |
| <b>Кожемякин П. Н., Парамонов А. И.</b><br>Программный комплекс контроля работы сердечно-сосудистой системы человека . . . . .                                | <b>204</b> |
| <b>Леченко А. В., Петровский Н. А.</b><br>Применение глубокого автоэнкодера для схемы сжатия изображения на основе параунитарного банка фильтров . . . . .    | <b>206</b> |

|                                                                                                                                                                       |            |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Мазура А. А., Гуринович А. Б.</b><br>Tree shaking как метод оптимизации сборки Web-приложения . . . . .                                                            | <b>208</b> |
| <b>Мазура И. А., Гуринович А. Б.</b><br>Извлечение ключевых слов: графоориентированный подход . . . . .                                                               | <b>210</b> |
| <b>Навроцкий А. А., Кривальцевич Е. В.</b><br>Автоматизированные диалоговые системы . . . . .                                                                         | <b>212</b> |
| <b>Оберемко М. И.</b><br>Алгоритмы и методы управления платежным агрегатором . . . . .                                                                                | <b>214</b> |
| <b>Паланевич А. С., Жук Е. Е.</b><br>Непараметрическая классификация многомерных наблюдений на основе ядерных оценок плотностей с прямоугольным ядром . . . . .       | <b>216</b> |
| <b>Папкович А. С.</b><br>Алгоритм регулирования цены в условиях неопределенности . . . . .                                                                            | <b>218</b> |
| <b>Плиско И. Г., Ганжин П. С., Ковалев М. А.</b><br>Информационная система учета научной работы сотрудников кафедры и прогнозирование рейтинга университета . . . . . | <b>220</b> |
| <b>Ревотюк М. П., Бебех А. В., Хаджинова Н. В.</b><br>Оценка устойчивости кратчайших путей на динамически определяемых графах . . . . .                               | <b>222</b> |
| <b>Серебряная Л. В., Брановицкий А. А.</b><br>Машинное распознавание эмоций по голосу . . . . .                                                                       | <b>224</b> |
| <b>Тиханович Т. В., Боброва Т. С.</b><br>Особенности подготовки специалистов в области информационных технологий . . . . .                                            | <b>226</b> |
| <b>Христофорова А. А., Гуринович А. Б.</b><br>Алгоритм загрузки многомерных данных в денормализованное хранилище . . . . .                                            | <b>228</b> |
| <b>Черемисинова Л. Д., Черемисинов Д. И.</b><br>Извлечение графа соединений компонент из описания СБИС на языке VHDL . . . . .                                        | <b>230</b> |
| <b>Шилин Л. Ю., Козарь Р. В., Навроцкий А. А.</b><br>Задача кластеризации при распознавании эндоскопических снимков . . . . .                                         | <b>232</b> |
| <b>Mukha V. S., Kako N. F.</b><br>Model-based optimization of the regression objects . . . . .                                                                        | <b>234</b> |
| <b>Nasr S.</b><br>Selecting answers from CV text . . . . .                                                                                                            | <b>236</b> |

*Научное издание*

**Информационные технологии и системы 2021  
(ИТС 2021)**

Материалы международной научной конференции  
(БГУИР, Минск, Беларусь, 24 ноября 2021)

**Information Technologies and Systems 2021  
(ITS 2021)**

Proceeding of The International Conference  
(BSUIR, Minsk, Belarus, 24th November 2021)

В авторской редакции

Ответственный за выпуск *Л. Ю. Шилин, А. Б. Гуринович*

Компьютерная верстка *Н. В. Гракова, А. Ф. Трофимович*

Дизайн обложки *А. А. Навроцкий*

Подписано в печать 15.11.2021. Формат 60x84 1/8. Бумага офсетная. Гарнитура «BookAntiqua».  
Отпечатано на ризографе. Усл. печ. л. 41,15. Уч.-изд. л. 45,3. Тираж 100 экз. Заказ 401.

Издатель и полиграфическое исполнение: учреждение образования  
«Белорусский государственный университет информатики и радиоэлектроники»  
ЛИ №02330/0494371 от 16.03.2009. ЛП №02330/0494175 от 03.04.2009.  
220013, Минск, П. Бровки, 6.