

**Министерство образования Республики Беларусь**  
Учреждение образования  
«Белорусский государственный университет  
информатики и радиоэлектроники»

## **КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ**

**56-я научная конференция  
аспирантов, магистрантов и студентов**

Сборник тезисов докладов

21–24 апреля 2020 года  
Минск, БГУИР

УДК 004.021+004.2+004.4+004.6+004.7+004.9  
ББК 32.97+32.81

56-я конференция аспирантов, магистрантов и студентов учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», 21-24 апреля 2020 г., БГУИР, Минск, Беларусь: тезисы докладов. – Мн. – 2020. – 241 с.; ил.

В сборнике опубликованы тезисы докладов, представленных на 56-й научной конференции аспирантов, магистрантов и студентов БГУИР. Материалы одобрены оргкомитетом и публикуются в авторской редакции.

Для научных и инженерно-технических работников, преподавателей, аспирантов, магистрантов и студентов вузов.

## СОДЕРЖАНИЕ

<b>СЕКЦИЯ «ЭЛЕКТРОННЫЕ ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ»</b>	<b>9</b>
1. ОБЗОР ХАРАКТЕРИСТИК И ПРИЗНАКОВ, ОПРЕДЕЛЯЮЩИХ ПОВЕДЕНИЕ ПОЛЬЗОВАТЕЛЯ ПК	10
2. ОБНОВЛЕНИЕ ПРОШИВКИ МИКРОКОНТРОЛЛЕРА ESP32 ПО ТЕХНОЛОГИИ ОТА	11
3. СКРЫТАЯ МАРКОВСКАЯ МОДЕЛЬ ДЛЯ ПРИНЯТИЯ РЕШЕНИЙ О КЛАССИФИКАЦИИ ВРЕМЕННОГО РЯДА СОБЫТИЙ В КОМПЬЮТЕРНОЙ СЕТИ	12
4. ИССЛЕДОВАНИЕ АЛГОРИТМА ЗАМЕЩЕНИЯ LAST LEVEL CACHE НА ПРИМЕРЕ ПРОЦЕССОРОВ INTEL	14
5. РАСПРЕДЕЛЕННАЯ СИСТЕМА РЕШЕНИЯ ЗАДАЧ	15
6. ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ ESP-NOW ДЛЯ ВЗАИМОДЕЙСТВИЯ РОЯ РОБОТОВ	17
7. ОБНАРУЖЕНИЕ И РАСПОЗНАВАНИЕ НОМЕРНЫХ ЗНАКОВ МЕТОДОМ ПОИСКА УНИКАЛЬНЫХ ГРАНИЦ	18
8. ОНЛАЙН-ПЛАТФОРМА ДЛЯ АНАЛИЗА И ОБРАБОТКИ ДАННЫХ	19
9. МУЛЬТИЗАДАЧНЫЙ РОБОТ С ФУНКЦИЕЙ СЛЕЖЕНИЯ ЗА ОБЪЕКТОМ	20
10. АТАКИ С ИСПОЛЬЗОВАНИЕМ ПРОТОКОЛА DNS И СРЕДСТВА ПРОТИВОДЕЙСТВИЯ ИМ	21
11. ВЗАИМОПОЗИЦИОНИРОВАНИЕ МОБИЛЬНЫХ РОБОТОВ С ПРИМЕНЕНИЕМ MESH-СЕТЕЙ	22
12. ИДЕНТИФИКАЦИЯ НЕИСПОЛЬЗУЕМЫХ ЗЕМЕЛЬ НА СНИМКАХ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ	24
13. МЕТОДИКА УСТРАНЕНИЯ ШУМА НА ОКТ-ИЗОБРАЖЕНИЯХ	26
14. ПРЕИМУЩЕСТВА ЭЛЕКТРОННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ФОРМИРОВАНИЕМ РЕЧЕВЫХ НАВЫКОВ ВЛАДЕНИЯ ИНОСТРАННЫМ ЯЗЫКОМ	28
15. АРХИТЕКТУРА РАСПРЕДЕЛЕННОЙ СИСТЕМЫ ОБРАБОТКИ ВИДЕО	30
16. ДИНАМИЧЕСКАЯ СИМУЛЯЦИЯ ОБЪЕМНОГО ОГНЯ	32
17. ПОДХОД К СОЕДИНЕНИЮ МНОЖЕСТВА КАМЕР С RASPBERRY PI ДЛЯ СЪЕМКИ СФЕРИЧЕСКИХ ПАНОРАМ	33
18. МОНИТОРИНГ И ПРОГНОЗИРОВАНИЕ УСТАЛОСТИ ВОДИТЕЛЯ ЗА РУЛЕМ	34
19. ВОССТАНОВЛЕНИЕ ИЗОБРАЖЕНИЙ С ПОМОЩЬЮ НЕОБУЧЕННЫХ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ	36
20. ИССЛЕДОВАНИЕ ПОДХОДОВ К РЕАЛИЗАЦИИ ОПЕРАЦИИ СВЁРТКИ НА ОСНОВЕ ТЕХНОЛОГИИ CUDA	38
21. ШУМОПОДАВЛЕНИЕ НА ИЗОБРАЖЕНИИ С ПОМОЩЬЮ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ	40
22. СЕМАНТИЧЕСКАЯ СЕГМЕНТАЦИЯ АЭРОФОТОСНИМКОВ СЕЛЬСКОХОЗЯЙСТВЕННОЙ РАСТИТЕЛЬНОСТИ НА БАЗЕ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ	42

23.	СИСТЕМА НАВИГАЦИИ МОБИЛЬНОГО РОБОТА	44
<b>СЕКЦИЯ «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»</b>		<b>46</b>
24.	МЕТОД ОБНАРУЖЕНИЯ И РАСПОЗНАВАНИЯ ДВИЖУЩИХСЯ ОБЪЕКТОВ В ВИДЕОПОТОКЕ	47
25.	СПОСОБЫ РАСЧЁТА ТРАЕКТОРИИ ДВИЖЕНИЯ ТРАНСПОРТА	48
26.	ПРОГРАММНОЕ СРЕДСТВО РЕАЛИЗАЦИИ ПРЕОБРАЗОВАНИЯ ГИЛЬБЕРТА-ХУАНГА	49
27.	АНАЛИЗ СТРУКТУР НЕЙРОННЫХ СЕТЕЙ ENCODER-DECODER ДЛЯ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ НА МОБИЛЬНЫХ УСТРОЙСТВАХ	51
28.	ПРИМЕНЕНИЕ МЕТОДА ПОИСКА КЛЮЧЕВЫХ СЛОВ ДЛЯ ЗАДАЧИ ИНДЕКСИРОВАНИЯ ДОКУМЕНТОВ	53
29.	ИССЛЕДОВАНИЕ СРЕДСТВ СОЗДАНИЯ И РЕАЛИЗАЦИИ ЭЛЕКТРОННЫХ ЦЕННЫХ БУМАГ КОМПАНИИ НА ОСНОВЕ ТЕХНОЛОГИИ БЛОКЧЕЙН	54
30.	ПРОГРАММНОЕ СРЕДСТВО ОЦЕНКИ ДЕМПФИРУЮЩИХ СВОЙСТВ ПОКРЫТИЙ	55
31.	МЕТОДЫ ОБФУСКАЦИИ ПРОГРАММНЫХ СРЕДСТВ С ЦЕЛЬЮ ЗАЩИТЫ ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ	57
32.	ПРОГРАММНОЕ СРЕДСТВО МОДЕЛИРОВАНИЯ И ПОИСКА НЕИСПРАВНОСТЕЙ ЗАПОМИНАЮЩИХ УСТРОЙСТВ	58
33.	ПРОГРАММНОЕ СРЕДСТВО АВТОМАТИЗАЦИИ ВЗВЕШИВАНИЯ ЖЕЛЕЗНОДОРОЖНЫХ ВАГОНОВ В ДИНАМИЧЕСКОМ РЕЖИМЕ	60
34.	ПРАКТИЧЕСКИЙ АНАЛИЗ ЭФФЕКТИВНОСТИ АЛГОРИТМОВ RATCHWORK И КОХА ДЛЯ РЕАЛИЗАЦИИ СИСТЕМЫ ВОДЯНЫХ ЗНАКОВ	61
35.	РАСПРОСТРАНЕНИЕ МЕДИАСООБЩЕНИЙ В СЕТИ ИНТЕРНЕТ	63
36.	ИСПОЛЬЗОВАНИЕ АЛГОРИТМА ДЕРЕВА РЕШЕНИЙ ДЛЯ АНАЛИЗА МНОГОМЕРНЫХ ДАННЫХ НА ПРИМЕРЕ ДАННЫХ ПО ОНКОЛОГИЧЕСКИМ ЗАБОЛЕВАНИЯМ ЛЁГКИХ	64
37.	РЕАЛИЗАЦИЯ ИНЕРЦИОННОСТИ ДИНАМИЧЕСКИХ ЭЛЕМЕНТОВ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА ПРОГРАММНЫХ СРЕДСТВ	66
38.	ОДНОПЛАТНЫЕ КОМПЬЮТЕРЫ В СИСТЕМАХ ПОЛУЧЕНИЯ ПОТОКОВЫХ АУДИОДААННЫХ	67
39.	РАЗРАБОТКА МОДЕЛИ КЛАССИФИКАЦИИ ПОЖАРНОЙ ОПАСНОСТИ В ЛЕСАХ	68
40.	АЛГОРИТМЫ ЭМИССИИ БАНКОВСКИХ КАРТ	70
41.	ЗАГРУЗКА И СОВМЕСТИМОСТЬ ОПЕРАЦИОННЫХ СИСТЕМ	71
42.	ПРОГРАММНОЕ СРЕДСТВО ДЛЯ ХРАНЕНИЯ И АНАЛИЗА РЕЗУЛЬТАТОВ СОЦИОЛОГИЧЕСКИХ ИССЛЕДОВАНИЙ С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ ASP.NET	73
43.	ВРМ СИСТЕМЫ КАК ФАКТОР ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ПРЕДПРИЯТИЯ	75
44.	ТЕНДЕНЦИИ РАЗВИТИЯ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ В БГУИР	76
45.	ФУНКЦИОНАЛЬНАЯ МОДЕЛЬ ИНФОРМАЦИОННОЙ СИСТЕМЫ АГРЕГАЦИИ БИЗНЕС-ДААННЫХ	80
46.	РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ СИСТЕМЫ 1С-BITRIX	82

47.	РАСШИРЕНИЕ МОДЕЛИ ОБНАРУЖЕНИЯ УЯЗВИМОСТЕЙ В WEB-ПРИЛОЖЕНИЯХ, ОСНОВАННОЙ НА СТАТИЧЕСКОМ АНАЛИЗЕ ИСХОДНЫХ КОДОВ	84
48.	ОБЗОР ВОЗМОЖНОСТЕЙ ПЛАТФОРМЫ FLUTTER	86
49.	ПРОГРАММНОЕ СРЕДСТВО СОХРАНЕНИЯ И ОБРАБОТКИ ТЕХНИЧЕСКИХ ПАРАМЕТРОВ ФРОНТАЛЬНОГО ПОГРУЗЧИКА НА ОСНОВЕ ТЕХНОЛОГИИ MICROSOFT AZURE	88
50.	ПРОГРАММНОЕ СРЕДСТВО СБОРА И АНАЛИЗА ТЕХНОЛОГИЧЕСКИХ ПАРАМЕТРОВ СТРЕЛЫ ФРОНТАЛЬНОГО ПОГРУЗЧИКА	89
51.	ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ ПОСТРОЕНИЯ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ	90
52.	ПОДХОД CQRS В WEB-ПРИЛОЖЕНИЯХ НА МОНОЛИТНОЙ АРХИТЕКТУРЕ	92
53.	АВТОМАТИЗАЦИЯ СОЗДАНИЯ ИГРОВЫХ ПРИЛОЖЕНИЙ ДЛЯ СОЦИАЛЬНОЙ СЕТИ	94
54.	ИСПОЛЬЗОВАНИЕ ДИНАМИЧЕСКОЙ СТРУКТУРЫ ИЗ КОНЕЧНЫХ АВТОМАТОВ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ПОИСКА ШАБЛОНА В ТЕКСТЕ	95
55.	АРХИТЕКТУРНОЕ РЕШЕНИЕ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В СФЕРЕ ТУРИЗМА	96
56.	ОБРАБОТКА БЛАНКОВ ОТВЕТОВ ДЛЯ АВТОМАТИЗАЦИИ ПРОВЕДЕНИЯ БУМАЖНОГО ТЕСТИРОВАНИЯ	97
57.	МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ ДЛЯ ВЫЯВЛЕНИЯ АНОМАЛИЙ В ПОСЛЕПОЛЕТНЫХ ДАННЫХ ОБЪЕКТИВНОГО КОНТРОЛЯ БПЛА	99
58.	ИНФОРМАЦИОННАЯ СИСТЕМА СБОРА И ОБРАБОТКИ ТЕЛЕМЕТРИЧЕСКИХ ДАННЫХ	100
59.	ОБЩИЕ ПОДХОДЫ К ПОСТРОЕНИЮ ЭМУЛЯТОРОВ X86-СОВМЕСТИМЫХ ПРОЦЕССОРОВ	102
60.	МЕТОДЫ ЗАЩИТЫ АВТОРСКОГО ПРАВА НА ПРОГРАММНЫЕ ПРОДУКТЫ С ПОМОЩЬЮ ВОДЯНЫХ ЗНАКОВ И ОТПЕЧАТКОВ ПАЛЬЦЕВ	103
61.	ПРОГРАММНОЕ СРЕДСТВО АНАЛИЗА РЕЛЬЕФА МЕСТНОСТИ НА ОСНОВЕ ДАННЫХ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ	104
62.	АНАЛИТИЧЕСКАЯ МОДЕЛЬ ТЕЧЕНИЯ ОНКОЛОГИЧЕСКИХ ЗАБОЛЕВАНИЙ	105
63.	VRM СИСТЕМЫ КАК ФАКТОР ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ПРЕДПРИЯТИЯ	106
64.	ПРОГРАММНОЕ СРЕДСТВО ФОРМИРОВАНИЯ ОБУЧАЮЩЕЙ ТАКТИКИ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННОЙ СЕТИ	107
	<b>СЕКЦИЯ «ВЫСШАЯ МАТЕМАТИКА»</b>	<b>109</b>
65.	МАТЕМАТИКА И МАТЕМАТИКИ В АНЕКДОТАХ И ШУТКАХ ЧАСТЬ 1, 2	110
66.	ОПТИМИЗАЦИЯ ПРОЦЕССА ДЕЛЕНИЯ ИГРОКОВ НА ТРЕНИРОВКАХ ПО БАСКЕТБОЛУ	110
67.	ПРИМЕНЕНИЕ МАТЕМАТИЧЕСКИХ МЕТОДОВ СТАТИСТИКИ В МЕДИЦИНЕ	111
68.	ЦЕПОЧКИ ОКРУЖНОСТЕЙ, ВПИСАННЫХ В ПАРАБОЛУ	113

69.	ОПТИМИЗАЦИЯ ПОТРЕБИТЕЛЬСКОГО ВЫБОРА СОВРЕМЕННЫХ СТУДЕНТОВ НА ОСНОВЕ МНОГОПРОДУКТОВОЙ ФУНКЦИИ ПОЛЕЗНОСТИ ТИПА КОББА-ДУГЛАСА	116
70.	НАТУРАЛЬНЫЕ ТОЧКИ ПОД КРИВОЙ	118
71.	РАВЕНСТВО ПАРСЕВАЛЯ	119
72.	НЕЕВКЛИДОВА ГЕОМЕТРИЯ, ТОПОЛОГИЯ	120
	<b>СЕКЦИЯ «ИНФОРМАТИКА»</b>	<b>122</b>
73.	АЛГОРИТМ ПОДСЧЁТА РАЦИОНАЛЬНЫХ ТОЧЕК ВЛИЗИ ПАРАБОЛЫ	123
74.	АЛГОРИТМ АВТОМАТИЧЕСКОГО ОПИСАНИЯ ИЗОБРАЖЕНИЙ	125
75.	ОБЗОР МЕТОДОВ ИДЕНТИФИКАЦИИ ФЛЕШ-НАКОПИТЕЛЕЙ С ИСПОЛЬЗОВАНИЕМ ФИЗИЧЕСКИ НЕКЛОНИРУЕМЫХ ФУНКЦИЙ	127
76.	ИСПОЛЬЗОВАНИЕ ГЕНЕРАТОРОВ КОНСТАНТ ДЛЯ ФОРМИРОВАНИЯ НЕОПТИМИЗИРУЕМЫХ VHDL ОПИСАНИЙ	129
77.	ИСПОЛЬЗОВАНИЕ СИСТЕМЫ КОМПЬЮТЕРНОЙ АЛГЕБРЫ MAPLE В ДИФФЕРЕНЦИАЛЬНОЙ ГЕОМЕТРИИ	131
78.	ДЕТЕКТИРОВАНИЕ ВОЗГОРАНИЙ ПО ВИДЕОПОТОКУ	133
79.	МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ КОНТРОЛЯ ПИТАНИЯ	134
80.	СИСТЕМА РЕКОМЕНДАЦИИ ВИДЕОКОНТЕНТА	136
81.	СВЕРТОЧНАЯ НЕЙРОННАЯ СЕТЬ	137
82.	ПРОГНОЗИРОВАНИЕ ПОЛОЖЕНИЯ СПУТНИКОВ	138
83.	ОБЪЕКТ DETECTION ДЛЯ АВТОМАТИЗАЦИИ ОБРАБОТКИ ДОКУМЕНТОВ	139
84.	ТРАНЗАКЦИИ БЛОКЧЕЙН СИСТЕМ: СОДЕРЖАНИЕ И ЖИЗНЕННЫЙ ЦИКЛ	141
85.	АНАЛИЗ АППАРАТУРНЫХ ЗАТРАТ НА РЕАЛИЗАЦИЮ ЦИФРОВОГО УСТРОЙСТВА ВСЕВОЗМОЖНЫХ ПЕРЕСТАНОВОК	142
86.	КРИПТОГРАФИЧЕСКИЕ АЛГОРИТМЫ ШИФРОВ ЗАМЕНЫ	144
87.	РЕАЛИЗАЦИЯ КРИПТОГРАФИЧЕСКИХ МЕТОДОВ НА ЯЗЫКЕ C++	146
88.	АВТОМАТИЗИРОВАННАЯ СИСТЕМА КРАУДСОРСИНГА ДАННЫХ	149
89.	МЕТОД ШНОРРА В КРИПТОГРАФИИ	151
90.	ОТСЛЕЖИВАНИЕ ЛИЦ НА ВИДЕО С ИСПОЛЬЗОВАНИЕМ КЛАССИЧЕСКИХ И НЕЙРОСЕТЕВЫХ ПОДХОДОВ В КОМПЬЮТЕРНОМ ЗРЕНИИ	152
91.	РАСПОЗНАВАНИЕ ЛИЦ НА ИЗОБРАЖЕНИЯХ ПЛОХОГО КАЧЕСТВА С ИСПОЛЬЗОВАНИЕМ НЕЙРОСЕТЕВЫХ АЛГОРИТМОВ	153
92.	ВИЗУАЛИЗАТОР ПРОСТРАНСТВЕННО-ВРЕМЕННЫХ ФУНКЦИЙ	155
93.	ОЦЕНКА КАЧЕСТВА УЧЕБНОГО МАТЕРИАЛА ПОСРЕДСТВОМ МЕТОДИК АНАЛИЗА ДАННЫХ	156
94.	НЕЙРОСЕТЕВАЯ СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЯ О ВЫДАЧЕ КРЕДИТА	158
95.	АНАЛИЗ ЭМОЦИЙ ЧЕЛОВЕКА ПО ПОРТРЕТНОМУ СНИМКУ	159
96.	ОБНАРУЖЕНИЕ РАКА ЛЁГКОГО ПО СНИМКАМ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ	161

97.	ИСКУССТВЕННЫЙ НЕЙРОН	162
98.	РЕКУРРЕНТНЫЕ НЕЙРОННЫЕ СЕТИ	163
99.	ОЦЕНКА ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ МОДЕЛЕЙ КЛАССИЧЕСКОГО МАШИННОГО ОБУЧЕНИЯ ДЛЯ АНАЛИЗА ТОНАЛЬНОСТИ РЕЧЕВЫХ СИГНАЛОВ ЧЕЛОВЕКА	164
100.	ОЦЕНКА ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ АНАЛИЗА ТОНАЛЬНОСТИ РЕЧЕВЫХ СИГНАЛОВ ЧЕЛОВЕКА	165
101.	ПРИЛОЖЕНИЕ ДЛЯ КОНТРОЛЯ ПРОИЗВОДИТЕЛЬНОСТИ РАБОЧИХ ГРУПП	166
102.	СИСТЕМА ПОДБОРА РАЗРАБОТЧИКОВ ДЛЯ ВЫПОЛНЕНИЯ ЗАДАЧ НА КРУПНЫХ ПРОЕКТАХ	167
103.	ИНТЕГРИРОВАНИЕ ВЫРАЖЕНИЙ СОДЕРЖАЩИХ ПРОИЗВЕДЕНИЯ ПОКАЗАТЕЛЬНЫХ И АЛГЕБРАИЧЕСКИХ ФУНКЦИЙ	169
104.	УЯЗВИМОСТЬ FLASH NAND	170
105.	РЕКУРРЕНТНАЯ НЕЙРОННАЯ СЕТЬ	172
106.	ИССЛЕДОВАНИЕ РАСПРОСТРАНЕНИЯ СИГНАЛОВ НА ПАРАМЕТРИЧЕСКОЙ МОДЕЛИ ФНФ ТИПА АРБИТР	173
	<b>СЕКЦИЯ «ФИЛОСОФИЯ, ЛОГИКА И ЖИЗНЬ»</b>	<b>176</b>
107.	ФИЛОСОФСКИЕ И ГУМАНИСТИЧЕСКИЕ АСПЕКТЫ МЕЦЕНАТСТВА КАК СОЦИАЛЬНОГО ЯВЛЕНИЯ	177
108.	К ВОПРОСУ ОБ ОПРЕДЕЛЕНИИ КРИТЕРИЕВ НАУЧНОГО ЗНАНИЯ	179
109.	ВЛИЯНИЕ ФЕНОМЕНА ТУЛЬПЫ НА СОВРЕМЕННУЮ ФИЛОСОФИЮ	180
110.	СУБЛИМАЦИЯ И ЦИВИЛИЗАЦИЯ В ПСИХОАНАЛИЗЕ	182
111.	ТРАНСГУМАНИЗМ. НА ПУТИ К ПОСТЧЕЛОВЕКУ	183
112.	ФИЛОСОФИЯ ВЫБОРА ИНДИВИДА	184
113.	ФИЛОСОФИЯ АЛЬБЕРА КАМЮ В РОМАНЕ ЧАКА ПАЛАНИКА «БОЙЦОВСКИЙ КЛУБ»	186
114.	«МЕМ» КАК СМЫСЛООБРАЗУЮЩИЙ ИНСТРУМЕНТ ИНФОРМАЦИОННОЙ КОММУНИКАЦИИ	188
115.	КОГНИТИВНЫЕ ИСКАЖЕНИЯ И ИХ ВЛИЯНИЕ НА ИНДИВИДУАЛЬНЫЙ ВЫБОР	189
116.	ТЕОРИЯ И ИСТОРИЯ ТРАНСГУМАНИЗМА	190
117.	СОЗДАНИЕ СТРАТЕГИЙ ПОВЕДЕНИЯ ЧЕЛОВЕКА ПРИ ПОМОЩИ ИНСТРУМЕНТОВ НЛП	192
118.	РАССУЖДЕНИЯ О СМЕРТИ И СТАРЕНИИ	194
119.	ПРОБЛЕМЫ ФЕМИНИЗМА В СОВРЕМЕННОМ ОБЩЕСТВЕ	196
120.	ЛЕЙБНИЦ И ОСНОВАНИЯ СИМВОЛИЧЕСКОЙ ЛОГИКИ	197
	<b>СЕКЦИЯ «ФИЛОСОФИЯ И МЕТОДОЛОГИЯ НАУКИ»</b>	<b>199</b>
121.	«ВОЛЯ К ЖИЗНИ» АРТУРА ШОПЕНГАУЭРА, ИЛИ ПРОЛЕГОМЕНЫ «ВОЛИ К ВЛАСТИ» ФРИДРИХА НИЦШЕ	200
122.	ЭСТЕТИЧЕСКИЕ ПАРАДИГМЫ В УСЛОВИЯХ ИНФОРМАЦИОННОГО ОБЩЕСТВА	201
	<b>СЕКЦИЯ «ФИЗИКА»</b>	<b>203</b>

123. ОБРАБОТКА ИЗОБРАЖЕНИЙ АТОМНО-СИЛОВОГО МИКРОСКОПА С ПОМОЩЬЮ МЕТОДОВ МАТЕМАТИЧЕСКОЙ СТАТИСТИКИ	204
124. АНАЛИЗ РЕЗУЛЬТАТОВ ПРИМЕНЕНИЯ ЭВРИСТИЧЕСКИХ ТЕХНОЛОГИЙ В ИЗУЧЕНИИ ФИЗИКИ НА ПОТОКЕ ВМСИС В БГУИР	206
125. ПРИМЕНЕНИЕ РЕНТГЕНОСТРУКТУРНОГО АНАЛИЗА ДЛЯ ИЗУЧЕНИЯ ГИДРОКСИДОВ В ФОЛЬГАХ СПЛАВОВ AL-CR	208
126. МОДЕЛИРОВАНИЕ ЭФФЕКТА ХОЛЛА В ПОЛУПРОВОДНИКАХ С ПОМОЩЬЮ СРЕДЫ MATLAB	210
<b>СЕКЦИЯ «ВСТРАИВАЕМЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ РЕАЛЬНОГО ВРЕМЕНИ»</b>	<b>212</b>
127. ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ РАЗРЕЖЕННОГО БПФ ДЛЯ ОБРАБОТКИ ЗВУКОВЫХ СИГНАЛОВ НА МОБИЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ ПЛАТФОРМАХ	213
128. РЕАЛИЗАЦИЯ ХЭШ ФУНКЦИЙ И СПОСОБЫ ИХ ВЫЧИСЛЕНИЯ	214
129. АДАПТИВНАЯ ФИЛЬТРАЦИЯ ПРИ ОБРАБОТКЕ СИГНАЛОВ В ТРАКТАХ РЛС	215
130. БАЗОВЫЕ ОСНОВЫ ПЕРЕДАЧИ СИГНАЛОВ НАВИГАЦИОННЫХ СПУТНИКОВ	217
131. ЭЛЛИПТИЧЕСКИЕ КРИВЫЕ В ЭЛЕКТРОННОЙ ЦИФРОВОЙ ПОДПИСИ	219
132. ФОРМИРОВАНИЕ ДИАГРАММЫ НАПРАВЛЕННОСТИ ЦИФРОВОЙ ФАЗИРОВАННОЙ АНТЕННОЙ РЕШЕТКИ	221
133. ВЫСОКОПРОИЗВОДИТЕЛЬНАЯ РЕАЛИЗАЦИЯ HMAC SHA-256 НА БАЗЕ FPGA	223
134. ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ СИСТЕМ СПУТНИКОВОЙ НАВИГАЦИИ	224
135. ДЕТЕКТОР ЗАИКАНИЯ НА ОСНОВЕ МАШИННОГО ОБУЧЕНИЯ	226
136. СИСТЕМА ОЦЕНКИ РЕЧЕВОЙ ФУНКЦИИ ДЛЯ ПАЦИЕНТОВ С БАС	227
137. ИСПОЛЬЗОВАНИЕ ИНФОРМАЦИИ О ВОЗРАСТНЫХ ИЗМЕНЕНИЯХ СЛУХА ПРИ МОДЕЛИРОВАНИИ НЕЙРОСЕНСОРНОЙ ТУГОУХОСТИ	229
138. ХАРАКТЕРИСТИКИ И ВИДЫ ВСТРАИВАЕМЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ	231
139. АППАРАТНАЯ РЕАЛИЗАЦИЯ АЛГОРИТМОВ ДЕКОМПОЗИЦИИ БУЛЕВЫХ ФУНКЦИЙ	232
140. ПРИМЕНЕНИЕ МОДЕЛИ ЦВИКЕРА ДЛЯ ОЦЕНКИ УРОВНЯ ГРОМКОСТИ ЗВУКОВОГО СИГНАЛА ДЛЯ ТУГОУХОГО ЧЕЛОВЕКА	233
141. АНАЛИЗ АУДИОСИГНАЛОВ С ПОМОЩЬЮ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ НА БАЗЕ МОБИЛЬНЫХ ПЛАТФОРМ	235
142. МОДУЛЬ ЗВУКОВЫХ ЭФФЕКТОВ ДЛЯ СТРУННЫХ ИНСТРУМЕНТОВ НА БАЗЕ FPGA	236
143. МЕТОДЫ РЕАЛИЗАЦИИ ДВУМЕРНОГО ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ	238



**СЕКЦИЯ «ЭЛЕКТРОННЫЕ ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ»**

# ОБЗОР ХАРАКТЕРИСТИК И ПРИЗНАКОВ, ОПРЕДЕЛЯЮЩИХ ПОВЕДЕНИЕ ПОЛЬЗОВАТЕЛЯ ПК

Байдун Д.Р.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Насуро Е.В. – к.т.н., доцент каф. ЭВМ

Проблема надежной аутентификации пользователя компьютерной системы актуальна во всех сферах человеческой деятельности. Особое место среди средств аутентификации занимают биометрические методы, основанные на уникальности биометрической информации каждого человека. В данной работе рассматриваются характеристики и признаки уникальные для каждого отдельного пользователя компьютерной системы.

Любая система требует надежной защиты, и если от попыток взлома через интернет имеется богатый спектр антивирусных программ, то от взлома физического нас защищают пароли. Но существует шанс, что злоумышленник смог узнать пароль и тогда компьютер и все его данные остаются полностью беззащитными. Однако, каждый человек уникален – имеет свой темперамент, привычки и навыки. В связи с этим, биометрические данные каждого пользователя компьютерной системой будут различаться, как и используемое периферийное оборудование. Далее представлены характерные признаки, основываясь на которых, можно составить биометрический профиль каждого пользователя.

Характеристики и признаки:

1. Распознавание лица. Используя веб-камеру, можно сравнить картинку с камеры с сохраненной ранее.
2. Пароль. Проверка подлинности введенного пароля, количество неправильных попыток, скорость ввода пароля.
3. Действия пользователя при запуске системы. Например, сочетание запущенных программ, открытие определенных файлов и т.п.
4. Сочетания клавиш. Используются ли пользователем сочетания клавиш (например, ctrl+c или ctrl+v), либо используется только мышь/тачпад для выполнения команд. Аналогично с использованием клавиш панели NumLock.
5. Среднее время активности. Например, если это рабочий компьютер и среднее время его работы приходится на промежуток времени с 09.00 до 18.00, то будет подозрительна активность данного устройства в нерабочее время.
6. Отключение алгоритмов защиты компьютера (антивирусные программы, брандмауэр и т.п.).
7. Активность пользователя. Подразумевается время активности пользователя как в отдельно взятых программах, так и в определенных областях программ (например, хранилище паролей в браузерах).
8. Работа с программным обеспечением. Количество одновременно открытых копий программ, количество загруженных файлов из интернета и область работы данных файлов.
9. Внешние периферийные устройства. Название устройства, активность, область применения, мониторинг файлов, измененных данной активностью.
10. Клавиатурный почерк. Под данным термином, подразумевается несколько признаков присущих каждому пользователю: количество ошибок при наборе, интервалы между нажатиями клавиш, время удержания клавиш, число перекрытий между клавишами, степень ритмичности при наборе, скорость набора.
11. Жесты. Определенные движения мышью на рабочем столе и в отдельных программах.
12. Работа с мышью/тачпадом. Скорость и ритм клика в каждой отдельно взятой программе.

Представленные выше характеристики и признаки можно собрать у каждого пользователя ПК и хранить наравне с паролями. Подобная система не затребует дополнительного оборудования. Однако, для более высокой точности идентификации пользователя и минимизации ложных срабатываний, необходимы дальнейшие исследования и сбор информации для создания системы приоритетов для каждого отдельно взятого признака, для поиска возможной синергии и диссинергии между отдельными характеристиками, а также для создания самой системы защиты.

**Список использованных источников:**

1. А.В. Соколов. Защита информации в распределенных корпоративных сетях и системах / Соколов А.В., Шаньгин В.Ф. М.: ДМК Пресс, 2002. – 656 с.
2. Технологии биоидентификации и биометрический рынок / А. Евангели // PC Week/RE 2003, № 7. -- С. 24—25.
3. А.В. Еременко. Идентификационный потенциал клавиатурного почерка с учетом параметров вибрации и силы нажатия клавиши/ А.В. Еременко и др.. М.: Синергия, 2017. - 16 с.

# ОБНОВЛЕНИЕ ПРОШИВКИ МИКРОКОНТРОЛЛЕРА ESP32 ПО ТЕХНОЛОГИИ ОТА

Брычиков Т.Д.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Понкратов Артём Андреевич – ассистент каф. ЭВМ

Цель – разработка кода для обновления прошивки микроконтроллера esp32 используя технологию over-the-air programming. Так же требуется обеспечить безопасность для канала передачи данных и идентификации типа и версии прошивки сервером. В результате работы была разработана система, в которой esp32 может обновить свою прошивку получая данные от сервера по защищенному каналу. В разработанную систему входит код как для esp32, так и для сервера.

В настоящее время такие сферы как роботы, умные дома, интернет вещей набирают большую популярность. Согласно данным из статьи [1], эти сферы продолжат так развиваться в течении нескольких ближайших лет. Из-за увеличения количества устройств, затруднительным становится обновление через прямое подключение, и всё более востребованным становится автоматическое и удалённое обновление сразу нескольких устройств.

Популярной платформой для данных сфер является ESP32 [2]. Это платформа, которая включает в себя двухъядерный микропроцессор с тактовой частотой 240МГц с 520Кб ОЗУ, 448Кб ПЗУ, а также встроенный модуль Bluetooth 4.2 и Wi-Fi 802.11n. Это устройство обладает низкой стоимостью, небольшими габаритами и малым энергопотреблением, 12-битным АЦП, 8-битным ЦАП, большим количеством линий ввода/вывода, аппаратной реализацией протоколов шифрования и проводной коммуникации.

ESP32 предоставляет возможность обновления устройства при помощи популярной технологии over-the-air programming (далее ota), которая предназначена для обновления исполняемого алгоритма удалённым способом. Ota позволяет устройству обновлять себя на основе данных, полученных во время собственной работы. На пример, способом получения данных для обновления может быть Wi-Fi или Bluetooth. Ota функции записывают новый образ прошивки на специальный раздел устройства, предназначенный исключительно для этого. Дальше, при перезагрузке, устройство использует этот раздел для запуска. Если после обновления и запуска новой прошивки произойдёт критическая ошибка, то устройство произведёт откат до прошлой версии.

В результате работы была разработана система, в которой микроконтроллер ESP32 может обновить свою прошивку, получая данные от сервера по защищенному каналу.

Так как при обновлении данные проходят по глобальной сети, то они находятся фактически в открытом доступе, в связи с чем злоумышленник может их изменить или перехватить. Для того чтобы безопасно передавать эти данные использовался HyperText Transfer Protocol Secure (далее https), этот протокол поддерживает SSL и TLS шифрование. При таком типе подключения клиент при запросе получает сертификат, который проверяется в центре сертификации чтобы определить кто его выдал. Если подключение проходит проверку подлинности, то клиент и сервер договариваются о секретном ключе, при помощи которого они шифруют и декодируют данные.

Сервер должен автоматически обслуживать большое количество устройств, которые могут выполнять разную работу, поэтому он должен их распознать. Для этого устройство посылает на сервер запрос, при этом в самом url указывается версия и тип прошивки, на что сервер может выслать отрицательный (в случае отсутствия более новой версии) или положительный ответ (с ссылкой на новую версию в сообщении).

Для реализации сервера был выбран язык Python и фреймворк Django. Для конфигурации запуска сервера использовался docker-compose и nginx. Для генерации самоподписанных сертификатов использовался openssl.

Модуль для обновления esp32 был написан на языке программирования C++. Код был написан на фреймворке esp-idf, который разработан специально для данной платформы производителем [3].

#### Список использованных источников:

1. Gartner Hype Cycle [Электронный ресурс]. – Режим доступа : <https://www.gartner.com/smarterwithgartner/5-trends-emerge-in-gartner-hype-cycle-for-emerging-technologies-2018/>.
2. Спецификация микроконтроллеров семейства ESP32 [Электронный ресурс]. – Режим доступа : [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf).
3. Программный интерфейс микроконтроллера ESP32 [Электронный ресурс]. – Режим доступа : <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/index.html>

1. Athency A. / Brain Tumor Detection and Classification in MRI Images Antony Athency, M.A. Ancy Brigit, K.A. Fathima, Raju Dilin, M.C. Binish // International Journal of Innovative Research in Science, Engineering and Technology. – 2017. – Vol. 6, № 5, P. 84-89.

# СКРЫТАЯ МАРКОВСКАЯ МОДЕЛЬ ДЛЯ ПРИНЯТИЯ РЕШЕНИЙ О КЛАССИФИКАЦИИ ВРЕМЕННОГО РЯДА СОБЫТИЙ В КОМПЬЮТЕРНОЙ СЕТИ

Бубнов Я.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Иванов Н.Н. – канд. физ.-мат. наук., доцент

В статье предлагается метод решения проблемы бинарной классификации событий, наблюдаемых в компьютерной сети, и представленных в виде временного ряда с помощью скрытой марковской модели. Метод позволяет принять решение об отнесении серии результатов индивидуальной классификации каждого события от зашумленных детекторов к одному из двух классов.

Обнаружение вредоносной активности узлов корпоративной компьютерной сети является одной из важных прикладных задач. Ее решение позволяет предотвратить целевые кибератаки, направленные как на отказ системы в целом, так и на кражу конфиденциальной информации. На практике данная задача решается с помощью детекторов [1, 2], анализирующих содержимое передаваемых через сеть пакетов, или детекторов занимающиеся сбором системной информации непосредственно с узлов сети. Собранная информация сохраняется в виде временных рядов в таких системах событийного мониторинга, как: Prometheus, Zabbix, Nagios [3]. Оперирование подобными системами предполагает установку допустимых диапазонов значений, при которых обеспечивается нормальное функционирование системы. При выходе анализируемых значений за пределы установленных границ, операторы получают уведомление о необходимости ручного вмешательства.

Как правило, допустимые границы выбираются опытным путем [4], к тому же, в некоторых случаях действия оператора строго регламентированы и поддаются автоматизации. Таким образом, ставится задача определить в какой момент времени требуется вмешательство операторов, для ее решения предлагаем использовать скрытую марковскую модель.

Рассмотрим частный случай детекторов, вычисляющий вероятность отнесения передаваемого сетевого пакета к вредоносному трафику, другими словами – бинарный классификатор. Тогда наблюдаемым событием  $Y_t \in \mathbf{I}$  является результат работы классификатора для монотонно возрастающего времени  $t \in \mathbf{Z}$ . Определим марковскую сеть, представленную на рисунке 1, где каждому наблюдаемому событию поставлено в соответствие два возможных состояния, или класса.

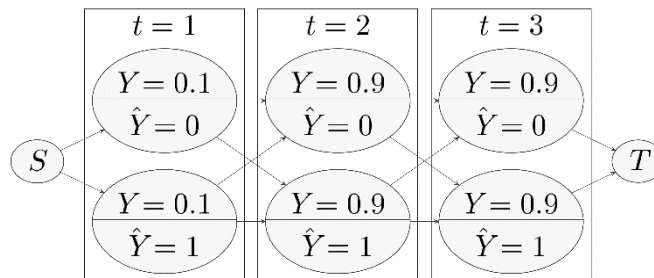


Рисунок 1 – Схема цепи Маркова серии наблюдаемых событий в компьютерной сети, где  $Y_t$  – вероятность отнесения наблюдения к одному из классов  $\hat{Y}$

Определим вероятность нахождения в конкретном состоянии  $i$  с помощью нормального распределения:

$$p_i = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{w}{\sigma}\right)^2}, \quad (1)$$

$$w = Y_i - \hat{Y}, \quad (2)$$

где среднеквадратическое отклонение составляет  $\sigma = \sqrt{1/5}$ . Другими словами, вероятнее всего скрытое состояние достоверно наблюдается, когда выход детектора находится в пределах интервала дисперсии эталонного класса.

Условимся, что наблюдаемые события образуют ординарный поток однородных событий, а значит могут быть описаны процессом Пуассона. Таким образом, определим вероятность перехода из состояния  $i$  в состояние  $j$  с помощью экспоненциального распределения:

$$p_{i,j} = e^{-z}, \quad (3)$$

$$z = \frac{t_i - t_j}{e^{1-Y_i \oplus Y_j}}, \quad (4)$$

где  $Y_i \oplus Y_j$  представляет собой полином Жегалкина – то есть, наиболее вероятны переходы из одинаковых классов, а переходы между различными классами дополнительно штрафуются.

Пусть задан некоторый интервал упорядоченных времен  $t \in \mathbf{T}$ , состоящий из  $N$  элементов, тогда задача определения результирующего класса события по наблюдениям в границах  $\mathbf{T}$  заключается в нахождении наиболее вероятного пути из состояния  $t_0$  в одно из состояний  $t_N$ .

Для поиска наиболее вероятного пути в цепи Маркова, представленной на рисунке 1, воспользуемся алгоритмом Витерби и определим следующие рекуррентные соотношения:

$$V_{1,n} = p_1, \quad (5)$$

$$V_{i,n} = \max_i^N (p_i \cdot p_{i,n} \cdot V_{i-1,i}). \quad (6)$$

Тогда скрытое конечное состояние определяется из уравнения:

$$x_N = \arg \max_i^N (V_{i,N}) \quad (7)$$

На самом деле нас в первую очередь интересует не реальные состояния цепи Маркова, а вероятность, что серия наблюдаемых событий относится к конкретному классу. Решение данной задачи напрямую: путем подсчета количества наблюдаемых классов в восстановленной последовательности, приводит к результату, который не учитывает временную локальность: события, произошедшие недавно в прошлом, влияют на результат больше, чем события из далекого прошлого.

Для решения описанной проблемы введем два виртуальных состояния в цепь Маркова, где два конечных состояния – равновероятные. Новая цепь Маркова представлена на рисунке 2.

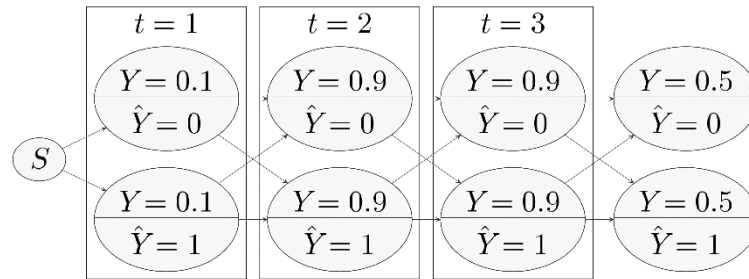


Рисунок 2 – Схема цепи Маркова серии наблюдаемых событий с равновероятными конечными состояниями

Вычислим вероятности нахождения в каждом из конечных состояниях с помощью алгоритма Витерби:

$$P_N^{\hat{Y}=0} = V_{t_N, N_0}, \quad (8)$$

$$P_N^{\hat{Y}=1} = V_{t_N, N_1}. \quad (9)$$

Тогда результирующий класс рассматриваемого временного ряда событий вычисляется по формуле:

$$Y = \arg \max (P_N^{\hat{Y}=0}, P_N^{\hat{Y}=1}) \quad (10)$$

Таким образом, в работе представлен бинарный классификатор временных рядов с обучением без учителя. Данный подход на практике позволяет принять решение о блокировке узла, при повторяющихся во времени событий с вредоносной активностью. Описанный метод при необходимости легко обобщается на многоклассовые задачи классификации.

**Список использованных источников:**

1. Qi, C. A bigram based real time DNS tunnel detection approach / C. Qi, X. Chen, C. Xu, J. Shi, P. Liu // Procedia Computer Science, Elsevier B.V. – 2013. – Vol. 17, P. 852-860.
2. Born, K. Detecting DNS Tunneling Using Character Frequency Analysis / K. Born, D. Gustafson // Proceedings of the 9th Annual Security Conference, Las Vegas 7-8 Apr 2010. – Las Vegas, 2010. - P. 2-3.
3. Skvortsov, P. Monitoring in the Clouds: Comparison of ECO2Clouds and EXCESS Monitoring Approaches / P. Skvortsov, D. Hoppe, A. Tenschert, M. Geinger // Proceedings of the 2nd International Workshop on Dynamic Resource Allocation and Management in Embedded, High Performance and Cloud Computing DREAMCloud, Prague 19 Jan 2016. – Prague, 2016. – P. 1-8.
4. Rong, K. ASAP: Prioritizing Attention via Time Series Smoothing / K. Rong, P. Bailis // Proceedings of Very Large Data Bases Endowment, Munich 28 Aug-1 Sep 2017. – Munich, 2017. – P. 1358-1369.

## ИССЛЕДОВАНИЕ АЛГОРИТМА ЗАМЕЩЕНИЯ LAST LEVEL CACHE НА ПРИМЕРЕ ПРОЦЕССОРОВ INTEL

Гагуа Д.Р., Кудырко А.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Перцев Д.Ю. – ст. преподаватель

Представлены проблемы, связанные с определением степени ассоциативности кэш-памяти процессора. Рассмотрены причины их появления и возможные способы их устранения.

В микроархитектуре x86 кэш играет существенную роль в ускорении взаимодействия АЛУ процессора и оперативной памяти. Типовой процессор включает иерархию уровней кэшей, каждый из которых построен на основе множественно-ассоциативной архитектуры. Зная, как организован доступ на каждом уровне кэша, возможно существенно ускорить вычисления. Например, код перемножения матриц большого объема при оптимизации с учетом правил доступа в кэш может быть ускорен более чем в 5 раз.

Основной способ определения степени ассоциативности кэш памяти сводится к следующему псевдокоду (Nmax - максимальная предполагаемая степень ассоциативности):

```
for (int N = 2; N < Nmax; N++) {
    1) init
    2) t = 0;
    for (int k = 0; k < 1000000; k++) {
        t = arr[t];
    }
}
```

Измеряя время работы шага 2 и сравнивая результат между собой, можно зафиксировать резкое увеличение времени работы цикла, т.е. “словить” время перехода между уровнями кэша либо выход в ОЗУ. Пример этапа инициализации init для N = 2 показан на рис.1.

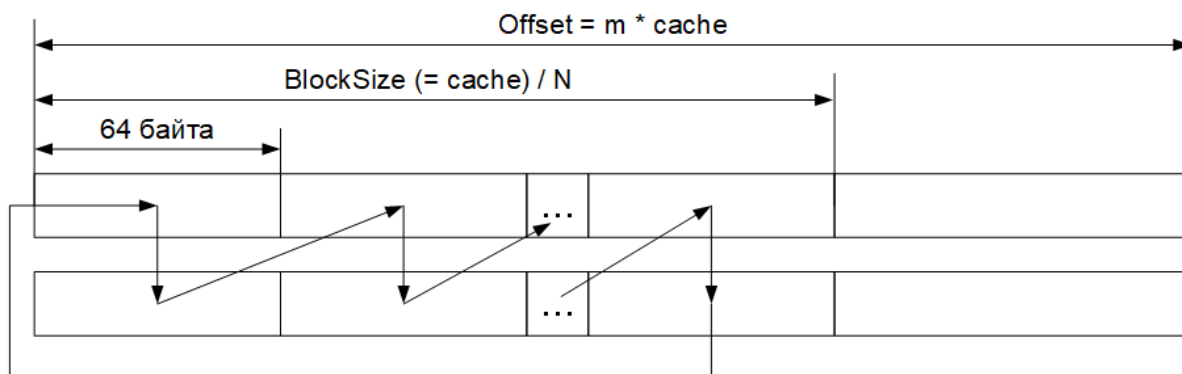


Рисунок 1 – Пример инициализации кэша

Однако проведенные эксперименты показывают, представленный алгоритм работает только при переходе между уровнями кэш-памяти и прекращает показывать корректные результаты при доступе в ОЗУ для большинства x86-процессоров, начиная с процессоров Intel Core2Duo. Это связано с тем, что начиная с данного поколения процессоров, компания Intel перешла на использование Smart Cache для последнего уровня (Last Level Cache) и схема организации изменилась (рис. 2).

Last Level Cache делится на фрагменты одинакового размера (slice). Предполагается, что данные из ОЗУ могут попасть в произвольный фрагмент, однако контроллер памяти "пытается" учитывать, для какого ядра предназначены данные и размещает их в максимально близкий к потребителю. С учетом этого, традиционный алгоритм замещения LRU (последний используемый) был замещен на хэш-функции, на основе которой вычисляется, в какой фрагмент и по какому адресу требуется разместить результат чтения памяти. Данное заключение, в том числе, подтверждается разными статьями [1-4]. Однако из-за того, что способ вычисления хэш-функции не документирован, такие изменения приводят к тому, что схема перемещения по памяти, описанная на рис. 1, перестает

работать. Статьи [1, 3] описывают подходы к анализу и способу вычисления хэш-функции.

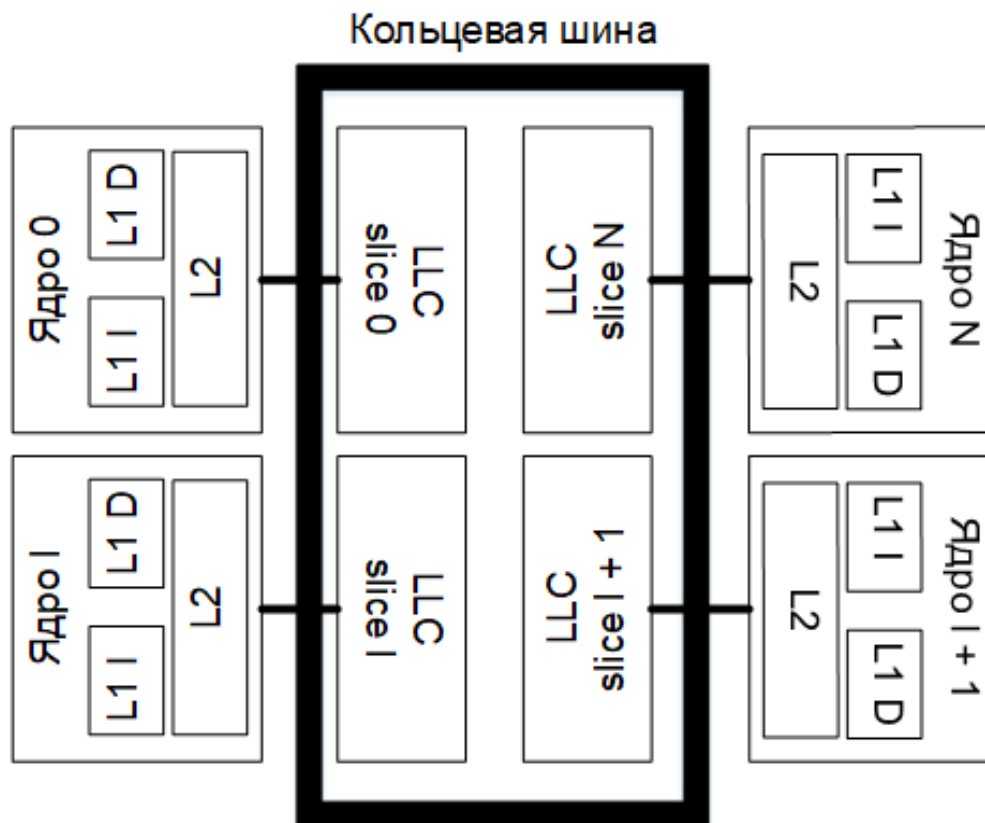


Рисунок 2 – Схема аппаратной реализации LLC процессора Intel

**Список использованных источников:**

1. Zhipeng W., Zehan C., Mingyu Ch. Cracking Intel Sandy Bridge's Cache Hash Function / W. Zhipeng [et al.]
2. Yuval Ya., Qian G., Fangfei L., Ruby B. L., Gernot H. Mapping the Intel Last-Level Cache / Ya Yuval [et al.]
3. Clémentine M., Nicolas Le S., Christoph N., Olivier H., Aurélien F. Reverse Engineering Intel Last-Level Cache Complex Addressing Using Performance Counters / M. Clémentine [et al.]
4. Farshin A., Roozbeh A., Kostić D., Maguire G. Q. Jr. Make the Most out of Last Level Cache in Intel Processors / A. Farshin [et al.]

## РАСПРЕДЕЛЕННАЯ СИСТЕМА РЕШЕНИЯ ЗАДАЧ

Гацура В.Д.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Матвейчук Н.М. – канд. физ.-мат. наук

В данной работе рассматривается проблема оптимального планирования распределенных задач. Разработана система, позволяющая составлять оптимальное или приближенное к оптимальному расписание выполнения задач, несмотря на неопределенность длительностей выполняемых работ.

Современные вычислительные нагрузки высоко изменчивы, и характеризуются особым распределением: множество небольших задач (или запросов) с малыми ресурсными требованиями и сравнительно небольшое множество крупных задач с непропорционально высокими требованиями к потребляемым ресурсам. При таких характеристиках задач становится актуальным планирование распределения задач по узлам обработки, которое позволит более эффективно использовать ресурсы.

В докладе рассматривается распределенная система решения задач для двух узлов, назовем их узел А и узел В. Имеется список задач, предназначенных к выполнению, среди которых можно выделить четыре группы задач: задачи, которые должны быть выполнены только на узле А; задачи, которые должны быть выполнены на узле В; задачи, которые должны быть выполнены сначала на узле А, а затем на узле В и задачи, которые должны быть выполнены сначала на узле В, а затем на

узле А. Для каждой задачи планируемая длительность ее выполнения на этапе составления расписания может быть известна лишь с определенной погрешностью. Поэтому вместо детерминированных длительностей операций будем рассматривать неопределенные (интервальные) длительности операций, предположив, что верхние и нижние границы будут заданы на этапе составления расписания.

Исходя из постановки задачи, можно сделать вывод что данную задачу можно рассматривать в терминах теории расписаний. Пусть для узлов А и В имеется список из  $n$  задач  $J = \{J_1, J_2, \dots, J_n\}$ . Фактическая длительность  $p_{ij}$  для каждой из задач не известна, но известны нижняя  $a_{ij}$  и верхняя  $b_{ij}$  границы:  $a_{ij} \leq p_{ij} \leq b_{ij}$ . В процессе выполнения возможные длительности задач принимают значения  $T = \{p \mid a_{ij} \leq p_{ij} \leq b_{ij}, J_i \in J, j \in \{A, B\}\}$  допустимых  $2n$ -мерных векторов  $p = (p_{1A}, p_{1B}, p_{2A}, \dots, p_{nB}) \in T$ .

Для того, чтобы рассматривать оптимальность или приближенность к оптимальности построенных расписаний, необходимо определить критерий, по которому и будет сделан вывод об оптимальности. Для данной задачи был выбран критерий минимизации общего времени выполнения заданного множества задач  $C_{max} = \max\{C_i \mid J_i \in J\}$ , где  $C_i$  – момент завершения выполнения задачи  $J_i \in J$ . Этот критерий является регулярным, поскольку целевая функция не убывает ни по одному из своих аргументов. Исходя из этого, оптимальное расписание достаточно искать среди активных расписаний, т. е. расписаний без неоправданных простоев каждого узла. Активное расписание задается перестановкой задач  $\pi_i = (J_{i1}, J_{i2}, \dots, J_{in}) \in S$ , определяющей порядок их выполнения. Здесь  $\{i_1, i_2, \dots, i_n\} = \{1, 2, \dots, n\}$  и  $S = \{\pi_1, \pi_2, \dots, \pi_n\}$  – множество  $n!$  перестановок  $n$  задач множества  $J$ . В обозначениях, принятых в теории расписаний, рассматриваемая неопределенная задача планирования времени двух узлов может быть представлена следующим образом:  $J2 \mid a_{ij} \leq p_{ij} \leq b_{ij}, n_i \leq 2 \mid C_{max}$ .

Оптимальное расписание для задачи  $J2 \mid C_{max}$  определяется парой перестановок Джексона  $(\pi', \pi'')$ , где  $\pi'$  – перестановка выполнения задач множества  $J_1 \cup J_{12} \cup J_{21}$  узлом А, а  $\pi''$  – перестановка выполнения задач множества  $J_2 \cup J_{12} \cup J_{21}$  узлом В. Решение задачи  $J2 \mid a_{ij} \leq p_{ij} \leq b_{ij}, n_i \leq 2 \mid C_{max}$  достаточно искать среди пар перестановок  $\pi' = (\pi'_{12}, \pi_1, \pi'_{21})$  и  $\pi'' = (\pi''_{21}, \pi_2, \pi''_{12})$ ,  $1 \leq i \leq n_{12}!$ ,  $1 \leq j \leq n_{21}!$  (задача  $J_j$  содержится в перестановке  $\pi_i$ , если  $J_j \in J_i, i \in \{1, 2, 12, 21\}$ ).

Поскольку длительность операций не определена, то в общем случае не существует перестановки, оптимальной для задачи  $J2 \mid a_{ij} \leq p_{ij} \leq b_{ij} \mid C_{max}$  при традиционном подходе к решению задачи (оптимальной для всех векторов длительностей операций  $p \in T$ ). Задача  $J2 \mid a_{ij} \leq p_{ij} \leq b_{ij} \mid C_{max}$  с математической точки зрения некорректна. Поэтому необходимо выполнять построение расписания в два этапа: до начала выполнения задач (этап off-line) и во время выполнения задач (этап on-line). На этапе off-line вначале производится попытка построения пары перестановок, оптимальных для всех векторов длительностей операций  $p \in T$ , для чего производится проверка достаточных условий. Если эти условия не выполняются, то на множестве требований производится построение графа частичного строгого порядка, что помогает сократить количество рассматриваемых перестановок. После этого, используя достаточные условия можно сделать вывод о возможности построения оптимального расписания на этапе off-line. Если эти достаточные условия не выполняются, то далее построение расписания будет производиться на этапе on-line. На этом этапе проверка достаточных условий производится в процессе обслуживания, когда операции имеют точные значения своих длительностей, выполнение которых уже завершено к этому моменту времени  $t$ . Знание точных значений длительностей операций позволяет уменьшить множество  $T$ . При проверке достаточных условий используются уже известные фактические значения времени выполнения завершенных задач. В случае невыполнения этих условий производится построение пары перестановок, оптимальных для средних значений времени выполнения задач.

Разработанная система позволяет составлять расписания выполнения задач узлами в два этапа, которые называются off-line и on-line. Также система во время своей работы собирает различные статистические данные построенных расписаний, которые в дальнейшем доступны для анализа. В случае, когда полученное расписание приближено к оптимальному, система дополнительно определяет оптимальное расписание с учетом фактических длительностей завершенных задач. На основании этого, в дальнейшем, рассчитывается разница во времени выполнения для того, чтобы оценить точность построенного расписания ввиду неопределенных длительностей. Для дальнейшего анализа в статистические данные также вносится информация о том, на каком этапе было составлено оптимальное расписание, количество времени, занимаемое программой для проведения подсчетов, количество простоев каждым из узлов. Время, занимаемое программой для проведения подсчетов, рассчитывается для того, чтобы узнать, насколько быстрым будет отклик программы в процессе составления расписания. На рисунке 1 представлена архитектура разработанной системы.

Компонент построения расписания на этапе off-line предназначен для построения первичного расписания выполнения задач. Если в результате работы данного компонента, расписание получилось оптимальным по достаточным условиям, то координатор отправляет задачи на выполнение без дальнейшего вмешательства в ход выполнения данных задач. В случае не



оптимальности расписания, координатор отправляет задачи на выполнение, но при этом отправляет команду компоненту построения расписания на этапе on-line отслеживать состояние выполнения задач и корректировать их ход по мере необходимости. Отслеживание состояния выполнения задач, а также их фактического времени выполнения производится компонентом мониторинга задач.

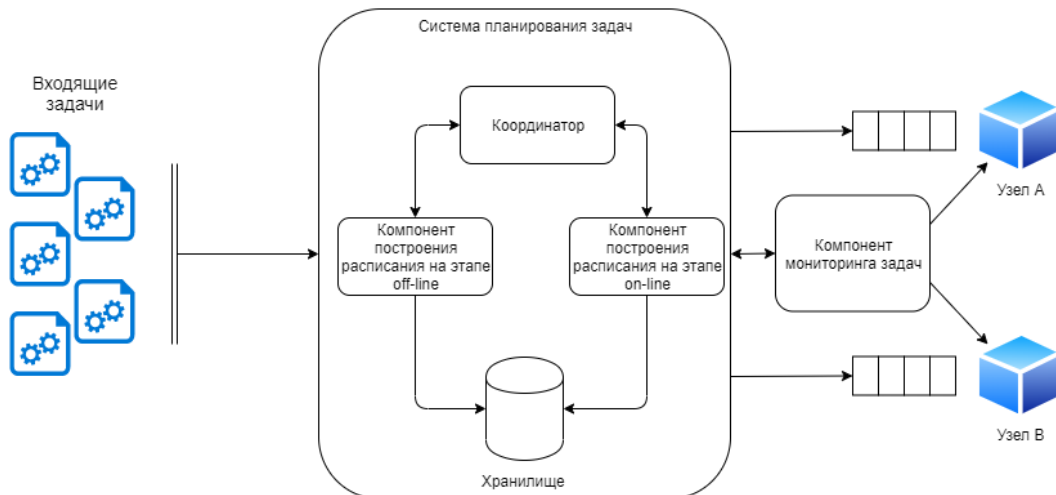


Рисунок 1 – Архитектура системы построения расписаний выполнения задач

К каждой задаче в списке входящих накладывается ограничение наличия следующей информации: границы длительности задачи, указание узла обработки данной задачи или последовательности узлов при необходимости выполнения задачи двумя узлами, вид разрешения конфликтов на этапе on-line. В результате выполнения входящих задач получаем построенное расписание вместе со статистическими данными по нему.

Основная архитектурная особенность разработки системы для построения расписания выполнения задач заключается в распределенности компонентов данной системы, что в свою очередь приводит к необходимости синхронизации работы компонентов для их слаженной работы. Механизмы синхронизации распределенных компонентов не является основной темой данного и доклада и по этой причине не рассматриваются нами.

**Список использованных источников:**

1. Sotskov, Yu.N. Two-machine job-shop scheduling problem to minimize the makespan with uncertain job durations / Yu.N. Sotskov, N.M. Matsveichuk, V.D. Hatsura // Algorithms. – 2020. – Vol. 13. №4. – P. 1-42.
2. Шафранский Я.М. О существовании глобально оптимальных расписаний для задачи Беллмана-Джонсона для двух приборов в условиях неопределенности / Я.М. Шафранский .. Информатика. – 2009. - №3. – С. 100-110.

## ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИИ ESP-NOW ДЛЯ ВЗАИМОДЕЙСТВИЯ РОЯ РОБОТОВ

*Гурская М. А., Гук В. А., Конигов А. Д., Понкратов А. А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Конигов А.Д. – ассистент кафедры ЭВМ*

Mesh-сети являются одноранговыми, что позволяет устанавливать соединение между участниками таких сетей без фиксированного центра управления этой сетью. Это открывает возможность создания одинаковых устройств, способных найти общение между собой в любом количестве. ESP-NOW – программный комплекс, разработанный создателем микроконтроллеров серии ESP, компанией Espressif. Он использует прямое взаимодействие между Wi-Fi.

С каждым днем сетевое взаимодействие становится все более важным. Кроме активного использования интернета людьми и персональными компьютерами, все большую долю трафика начинают использовать IoT решения. В виду этого, компания Espressif предлагает микроконтроллеры, направленные на взаимодействие по сети со встроенными модулями Wi-Fi и Bluetooth.

Передавать данные в распределенных системах на микроконтроллерах по Wi-Fi всегда было довольно сложно в виду того, что было необходимо подключаться к сети Wi-Fi, созданной маршрутизатором, что не всегда удобно при использовании аппаратных комплексов, управляемых

микроконтроллером.

Одним из решений является использование одного из устройств в качестве точки доступа (AP – access point), создающего сеть и контролирующего обмен данными между другими устройствами, работающими в режиме станции (STA - station). Но это решение имеет множество недостатков, так как делает невозможным общение между всеми одинаковыми устройствами. Кроме того, сеть сильно «привязывается» к одному устройству, выход которого из строя делает невозможным дальнейшее взаимодействие остальных участников сети. Еще одной проблемой является радиус действия сети устройства, выступающего в роли точки доступа, который ограничивает расположение участников сети территориально и поднимает вопрос о использовании ретрансляторов или введении многоуровневой сети, что значительно усложняет взаимодействие между ее участниками [1].

Выходом из этой ситуации является использование Mesh-сети. Mesh-сеть – это распределенная, одноранговая, ячеистая сеть. Каждый узел этой сети равнозначен по отношению к другим узлам, что делает возможным использование одного и того же программного обеспечения для управления всеми узлами данной сети.

Mesh-сети бывают настраиваемыми, когда они нуждаются в ручной настройке, и самоорганизующимися, когда новое оборудование, включенное в сеть, само подключается к существующим участникам, прокладывает маршруты, и настраивает себя в соответствии с сетью.

В виду сложности взаимодействия с устройствами, созданными на базе микроконтроллеров, учитывая минимальное количество и простоту средств ввода-вывода, необходимо создать самоорганизующуюся сеть, которая легко и быстро будет создаваться и настраиваться между любым количеством устройств.

Хорошей основой для подобной сети станет протокол ESP-NOW, позволяющий установить соединение между устройствами, построенными на базе микроконтроллеров серии ESP, производства компании Espressif, предоставляющие все необходимые аппаратные средства для создания подобных сетей [2].

В рамках этой технологии возможно реализовать unicast и broadcast передачу данных. В первом случае обращение происходит к конкретному устройству, а во втором ко всем участникам сети.

При передаче данных допускается возможность отправлять информацию, зашифрованную по протоколу CCMP, описанному в стандарте IEEE 802.11-2012 с использованием шестнадцатитбитных ключей PMK (Primary Master Key) и LMK (Local Master Key). PMK используется для шифрования LMK алгоритмом AES-128. Если PMK не задан, используется PMK по умолчанию. Шифрование поддерживается только для Unicast соединений [3].

Задача данной работы – создание самонастраиваемой сети для взаимодействия роя роботов, используя все особенности и возможности протокола ESP-NOW.

**Список использованных источников:**

1. Технологии современных беспроводных сетей Wi-Fi : учебное пособие / [Е. В. Смирнова, А. В. Пролетарский и др.] ; под общ. ред. А. В. Пролетарского. — Москва : Издательство МГТУ им. Н.Э. Баумана, 2017. — 446, [2] с. : ил. — (Компьютерные системы и сети).
2. ESP32 Series Datasheet [Электронный ресурс] : Datasheet / Espressif Systems – Режим доступа : [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf)
3. ESP-NOW User Guide [Электронный ресурс] : User Guide / Espressif Systems – Режим доступа : [https://www.espressif.com/sites/default/files/documentation/esp-now\\_user\\_guide\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp-now_user_guide_en.pdf)

## **ОБНАРУЖЕНИЕ И РАСПОЗНАВАНИЕ НОМЕРНЫХ ЗНАКОВ МЕТОДОМ ПОИСКА УНИКАЛЬНЫХ ГРАНИЦ**

*Завадский В.Г.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Воронов А.А. – к.т.н., доцент*

В настоящее время компьютерные технологии проникли практически во все сферы жизнедеятельности человека. Многие процессы и задачи, которые до недавнего времени возлагались на человека, сейчас полностью автоматизированы и практически не требуют вмешательства человека. Целью данной работы является исследование современных методов распознавания номерных знаков транспортных средств.

В алгоритме распознавания номерных знаков методом поиска уникальных границ можно выделить четыре основных этапа:

- 1) Предобработка;
- 2) Выделение области с номерным знаком;

- 3) Сегментация символов номерного знака;
- 4) Распознавание символов.

На этапе *предобработки* входное изображение полученное в RGB формате переводится в оттенки серого. Далее к полученному изображению применяется медианный фильтр. Таким образом с изображением будут убраны различные шумы. Так же побочным эффектом данного фильтра является выделение высокочастотных компонент, что будет полезно на следующих этапах распознавания [1].

Целью этапа *выделения области с номерным знаком* является локализация положения номерного знака на исходном изображении, так же получение изображения, которое содержит только символы номерного знака [2]. Данная процедура производится с помощью усредненного фильтра. Далее изображение бинаризуется и к нему применяется оператор Собеля для выделения точных границ бинаризованного изображения. Далее необходимо выделить непрерывную прямоугольную область по ширине и высоте пропорциональную размерам номерного знака.

На этапе *сегментации символов* полученная область с номерным знаком разбивается на отдельные символы, которые позже будут распознаны. Для выполнения данной задачи изображение бинаризуется с довольно низким порогом (порядка 0.01) [3-4]. Далее полученные области следует отфильтровать по размерам: нас интересуют области размером от 1000 до 8000 пикселей. Остальные области мы считаем различного рода мусором (винты крепления номера, грязь и тд). Для дальнейшего удобства распознавания каждое изображение с символом приводится к одному и тому же размеру, например 175x730 пикселей.

На этапе непосредственно *распознавания символов* мы на выходе получаем символьную строку строку с номерным знаком. Распознавание символов производится с помощью нейронной сети. Основным преимуществом данного метода является высокая точность при распознавании изображения. В данном конкретном примере используется нейронная сеть вида FFNN (feedforward neural network) [5]. Для обучения нейронной сети используется база Extended MNIST, которая включает более 60000 изображений символов и цифр.

**Список использованных источников:**

1. Мурыгин К.В. - Нормализация изображения автомобильного номера и сегментация символов для последующего распознавания // Институт проблем искусственного интеллекта МОН Украины и НАН Украины, г. Донецк 2010 г. - 6с.
2. Мурыгин К.В. Обнаружение автомобильных номеров на основе смешанного каскада классификаторов / К.В. Мурыгин // Искусственный интеллект. – 2010. – No 2. – С. 147-152.
3. Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс – М.: Техносфера, 2006. – 1072с.
4. Шередеко, Ю.Л. Способ корректного сведения задачи идентификации к задаче распознавания образов / Шередеко Ю.Л., Марусяк А.В. – УсиМ., 2002. – No5. – С.5- 12
5. Волкова, В.Н. Теория систем и системный анализ / Волкова В.Н. - М.: Издательство Юрайт, 2010.

## ОНЛАЙН-ПЛАТФОРМА ДЛЯ АНАЛИЗА И ОБРАБОТКИ ДАННЫХ

*Климец А.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Куприянова Д.В. – ассистент*

Представлен прототип разрабатываемой системы в рамках концепции Data Mining as a Service, т.е. предоставление сервиса по анализу и обработке данных, доступ к которому можно получить через браузер. Преимуществами системы является: ориентированность на клиент-серверную архитектуру, взаимодействие с любой библиотекой алгоритмов обработки данных, предоставляющей доступ с помощью языка программирования Python.

Объём генерируемой каждый день информации существенно увеличивается каждый день (например, по результатам исследований, проводимых в 2019 году в компании Domo [1], в Instagram создается 277 777 текстовых сообщений, а Google обрабатывает 4 497 420 поисковых запросов ежеминутно). Все создаваемые информационные потоки анализируются, обрабатываются и учитываются в различных сферах деятельности (реклама, прогноз успеха той или иной кампании). Так как «вручную» обработать весь поток информации невозможно, огромными темпами развивается такое направление, как Data Mining.

Data Mining – исследование и обнаружение в сырых данных скрытых знаний, которые ранее не были известны, нетривиальны, практически полезны, доступны для интерпретации человеком [2]. В данной области широкое развитие получили как коммерческие (например, MatLab, Statistica), так и свободные (например, Weka, R и др.) специализированные инструменты. Однако они обладают как преимуществами («заточка» на решение той или иной задачи), так и недостатками (написание кода, усложненный интерфейс со множеством параметров, работа только на персональном компьютере).

Разрабатываемая система предоставляет простой и удобный интерфейс пользователя, не

требующий какого-либо навыка в области написания программного кода или предварительного изучения того, как следует взаимодействовать с системой. В ней реализуются обширные возможности для расширения за счёт использования принципа модульности: можно подключить любую библиотеку на языке Python, который является главным языком программирования, используемый в проектах, связанных с математической статистикой, машинным обучением, нейронными сетями, анализом данных и т.д.

Система состоит из двух частей: сервера, предоставляющего API для взаимодействия с библиотеками языка Python и формирования конечной программы, которая будет осуществлять работу с данными, и веб-интерфейса, доступ к которому можно получить посредством сети Интернет. Интерфейс состоит из двух областей: списка доступных для использования функций, представляемого в виде многоуровневого списка, и области, в которой выстраивается порядок выполнения операций над данными, представленного в виде отдельных вкладок для каждой функции. Макет интерфейса пользователя приведён на рисунке 1.

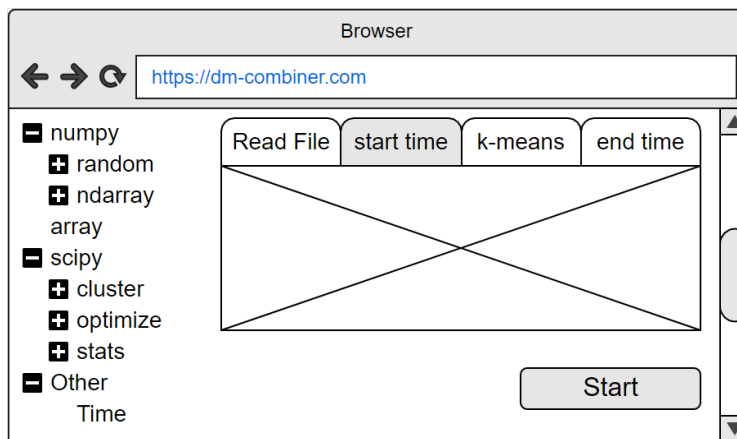


Рисунок 1 – Макет интерфейса пользователя системы

**Список использованных источников:**

1. Data Never Sleeps 7.0 [Электронный ресурс].– Режим доступа: <https://www.domo.com/learn/data-never-sleeps-7>.– Дата доступа: 25.03.2020
2. Барсегян, А.А. Анализ данных и процессов: учеб. пособие / А.А. Барсегян, М.С. Куприянов, И.И. Холод, М.Д. Тесс, С.И. Елизаров. – 3-е изд., перераб. и доп.– СПб.: БХВ-Петербург, 2009. – 512 с.

## МУЛЬТИЗАДАЧНЫЙ РОБОТ С ФУНКЦИЕЙ СЛЕЖЕНИЯ ЗА ОБЪЕКТОМ

*Ковбаса Г.А., Юревич В.С., Стракович А.И., Гук В.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Перцев Д.Ю. – старший преподаватель*

Представлено концептуальное описание мультизадачного робота с функцией слежения за объектом, его основной функционал, используемые программно-аппаратные средства.

Все больше компаний представляют на рынок свои варианты роботов-ассистентов для широкого круга пользователей. Такие устройства различаются в возможностях и себестоимости, начиная от более простых (умных колонок от компаний Яндекс, Microsoft) и заканчивая персональными роботами для домашнего использования, такими как Aido, Kuri, Buddy[1]. Эти устройства служат различным целям: от обучения до повышения эффективности работы систем «умный дом», снижая контроль со стороны пользователя до минимума. Персональные помощники роботов оказываются также полезными для пожилых или людей с ограниченными возможностями. К примеру, Assistant Personal Robot (APR) – это роботизированная платформа с дистанционным управлением, созданная для предоставления социальных и вспомогательных услуг людям с умеренными когнитивными нарушениями.

В данной работе представлен мультизадачный робот-ассистент для офисных и жилых помещений. В рамках работы над устройством выделены основные аспекты его текущего[2] и планируемого функционала, такие как:

- распознавание речи и обработка команд посредством системы Google;
- настройка и управление состоянием посредством андроид-приложения;
- независимое перемещение по помещению, избегая возможные препятствия;
- распознавание лиц и образов в реальном времени, слежение за распознанным объектом.

Данный робот основан на микрокомпьютере Raspberry Pi 4B с 4 Гб оперативной памяти. Это позволяет повысить уровень распознавания за счет возможности съемки видео в большем разрешении при сохраняемой высокой скорости работы. Сервис на языке Python управляет как съемкой и обработкой изображения для распознавания образов, так и управлением движения за счет передачи команд через эмуляцию COM-порта на Arduino Uno. К Arduino подключены драйвера двигателей, сервоприводов, ИК и ультразвуковые датчики. Такое решение позволяет освободить микрокомпьютер от необходимости выделения процессорного времени на обращение к датчикам и двигателям для перемещения в пространстве.

В данный момент для распознавания человеческого лица применяется комбинация методов Хаара и LBPH [3], обучение модуля распознавания проводится средствами OpenCV[4]. Данный метод прост и эффективен для маломощных систем при правильном подходе, так как не требует большой вычислительной мощности и не занимает много времени на обучение классификатора. Основная сложность в данном подходе – недостаточная точность распознавателя. Для решения данной проблемы применяется предобработка полученных изображений с видеопотока. Основными шагами предобработки являются: перевод в градации серого, нормализация освещенности, удаление шумов, повышение контрастности полученного снимка лица.

Кроме обработки видеопотока и взаимодействия с платформой Arduino, сервис также поддерживает передачу данных по Bluetooth. Клиентское приложение, спроектированное для платформы Android, позволяет подключиться к роботу посредством Bluetooth для установки WiFi-соединения и передачи команд для настройки распознавания лиц: добавление, удаление лиц из базы и переобучение распознавателя.

Мультифункциональный робот может применяться для удаленного контроля за состоянием помещений, в качестве интерактивного голосового ассистента, независимой охранной системы, а также как платформа для обучения учащихся и студентов аппаратному проектированию и программированию для Arduino и Raspberry.

#### **Список использованных источников:**

1. Информационный портал Top 5 robot personal assistants for your home in 2020 [Электронный ресурс]. Режим доступа: <https://roboticsbiz.com/top-5-best-robot-personal-assistants-for-your-home-in-2019/>. – Дата доступа: 4.03.2020.
2. 55-я юбилейная научная конференция аспирантов, магистрантов и студентов. Сборник тезисов докладов. [Электронный ресурс]. Режим доступа: [https://www.bsuir.by/m/12\\_100229\\_1\\_136895.pdf](https://www.bsuir.by/m/12_100229_1_136895.pdf). – Дата доступа: 10.03.2020. С. 42-44.
3. Применение локальных бинарных шаблонов к решению задачи распознавания лиц [Электронный ресурс]. Режим доступа: <https://habr.com/ru/post/193658/>. – Дата доступа: 10.02.2020.
4. Распознаем лица на фото с помощью Python и OpenCV [Электронный ресурс]. Режим доступа: <https://habr.com/ru/post/301096/>. – Дата доступа: 8.02.2020.

## **АТАКИ С ИСПОЛЬЗОВАНИЕМ ПРОТОКОЛА DNS И СРЕДСТВА ПРОТИВОДЕЙСТВИЯ ИМ**

*Коляго Н.Р.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Одинец Д.Н. – канд. техн. наук*

Данная работа содержит описание самых распространенных атак как на саму инфраструктуру DNS, так и с использованием архитектуры DNS, а также описывает существующие способы им противодействовать.

Протокол DNS разрабатывался с идеей, что инфраструктурой может пользоваться кто угодно, никакой аутентификации пользователя или шифрования данных не предусмотрено. Кроме того, он реализован с помощью протокола UDP, что позволяет более быстро обмениваться данными. Однако, именно эти особенности делают возможными большинство типов атак как на сам DNS, так и с использованием DNS.

Атаки с использованием инфраструктуры DNS в основном относятся к DoS атакам. К ним можно отнести «отраженные» атаки и «отраженные» атаки с усилением. Злоумышленник посылает на DNS сервер множество DNS запросов, в которых подменен IP адрес источника на IP адрес жертвы. В ответ сервер посылает множество ответов в адрес жертвы, чем вызывает трату ее ресурсов на обработку пришедших пакетов. Усиление же заключается в том, что запросы подбираются так, чтобы

ответ был значительно больше, чем сам запрос. Это вызывает еще большую трату ресурсов.

Эффективных способов защититься от таких атак нет, даже при настроенном фаерволле интернет-канал до него все равно будет занят мусорными пакетами, будут тратиться вычислительные ресурсы на отсеивание пакетов, а также придется заблокировать функционал DNS в целом.

Кроме DoS атак с помощью DNS можно произвести разведывательную атаку. Обычно, она является частью какой-то другой, более долгосрочной атаки. Например, с помощью DNS возможно получить информацию, на каком конкретно программном обеспечении работает сервер, а затем использовать уже специфичную для этого программного обеспечения атаку. Сервисы, позволяющие получать такую информацию, находятся в открытом доступе в сети Интернет.

Защититься от разведывательных атак можно только не допуская хранения чувствительной информации в системе DNS.

Самыми известными следствиями отсутствия какой-либо проверки авторитетности источников DNS пакетов являются атаки неавторизованного обновления и отравление кэша.

Обновление записей в реальном времени или по запросу является основной функцией DDNS (Dynamic DNS). Определенный тип запроса позволяет добавить или удалить ресурсную запись. Однако, изначальный механизм проверки источника был очень простым: IP адрес сверялся с хранящимся на сервере списком авторитетных источников. Злоумышленник, подменив адрес в пакете, мог добавить на сервер новую запись либо обновить существующую.

Отравление кэша (атака Каминского)[1] является атакой на рекурсивный сервер. Злоумышленник запрашивает у сервера несуществующий поддомен домена, который будет подменен. Сервер отправляет запрос на авторитетный сервер. Злоумышленник посылает на рекурсивный сервер множество подделанных ответов. С некоторой вероятностью один из них дойдет раньше реального ответа и будет принят. Пользователь, запросив IP подмененного хоста, получит тот IP, который задал злоумышленник.

Кроме этого, DNS трафик может быть модифицирован налету Интернет-провайдером.

Для защиты от таких атак были разработаны расширения DNSSEC (только аутентификация источника и проверка целостности, отсутствие шифрования данных для обеспечения обратной совместимости) и экспериментальный протокол DoH (DNS over HTTPS), который обеспечивает шифрование данных при передаче запросов и ответов между звеньями цепочки DNS.

Также, DNS-серверы уязвимы перед DoS атаками случайного поддомена и NXDOMAIN атаками. Они заключаются в посыле на сервер большого числа запросов на несуществующие поддомены (авторитетный сервер вынужден осуществлять более глубокий поиск соответствия) или несуществующие домены (рекурсивный сервер вынужден обращаться к авторитетному). При этом ресурсы сервера тратятся на заведомо бесполезный поиск соответствия. От таких атак невозможно полностью защититься, можно лишь смягчить последствия путем оптимизации программного кода сервера и фильтрации запросов.

**Список использованных источников:**

1. S. Son, The Hitchhiker's Guide to DNS Cache Poisoning [Electronic resource] / S. Son, V. Shmatikov // Cornell University. – Mode of access: [https://www.cs.cornell.edu/~shmat/shmat\\_securecomm10.pdf](https://www.cs.cornell.edu/~shmat/shmat_securecomm10.pdf). – Date of access: 15.03.2020
2. DNS Amplification Attack Detection and Mitigation via sFlow with Security-Centric SDN/ A. Atan [et al.] // IMCOM '17, Beppu, Japan, January, 2017 – P. 1–7.

## **ВЗАИМОПОЗИЦИОНИРОВАНИЕ МОБИЛЬНЫХ РОБОТОВ С ПРИМЕНЕНИЕМ MESH-СЕТЕЙ**

*Коников А.Д., Понкратов А.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Самаль Д.И. – канд. техн. наук, доцент*

В настоящее время набирают популярность группы автономных мобильных роботов, образующий рой. Соответственно, для координирования действий группы, каждому участнику роя необходимо знать, положение других относительно себя. Для решения этой задачи, в рамках данной исследовательской задачи, применяются алгоритмы триангуляции. Для обмена данными о цели и взаимном расположении участников роя применяется Mesh-сеть на основе технологии wi-fi.

Область автономных мобильных роботов набирает высокую популярность и получает всё большее развитие. По заслуживающим внимания оценкам, наибольшее развитие она получит в течение ближайших 5-10 лет [1]. Одним из малоисследованных направлений в данной области являются исследования коллективного интеллекта, иными словами роевого поведения. Рой – группа роботов с низким уровнем способностей, но формирующих в совокупности робота с высоким уровнем способностей. Множество независимых компактных роботов смогут объединяться для решения одной

большой и сложной задачи, непосильной для решения одним роботом либо требующей слишком много ресурсов: энергии, времени и т.д.[2] Примером такой задачи является перемещение крупного и тяжёлого объекта, либо большого множества маленьких объектов.

В рамках магистерской диссертации проводится исследование и разработка робототехнической платформы для исследования алгоритмов роевого поведения на основе пьезоробота.

При разработке и реализации группы автономных мобильных роботов, с целью исследования алгоритмов роевого поведения, возникает необходимость в том, чтобы каждый участник роя знал о расположении своих соседей. Для определения взаимного расположения мобильных роботов применяются алгоритмы триангуляции.

В качестве системы управления роботом, в рамках диссертации, используется платформа ESP32 [3], которая представляет собой двухядерный 32-битный микропроцессор с таковой частотой до 240МГц с 520Кб ОЗУ, 448Кб ПЗУ, а также имеет встроенный модуль Wi-Fi 802.11n и Bluetooth версии 4.2. Данный модуль обладает низкой стоимостью и малым энергопотреблением, большим количеством линий ввода/вывода – 34, плюс ко всему имеется 12-битный аналого-цифровой преобразователь, 8-битный цифро-аналоговый и множество других функций. К ним относятся 4 порта SPI, 2 порта I2C, 3 порта UART, 16 каналов ШИМ, инфракрасный порт, поддержка CAN шины, контроллер управления картами памяти и многое другое.

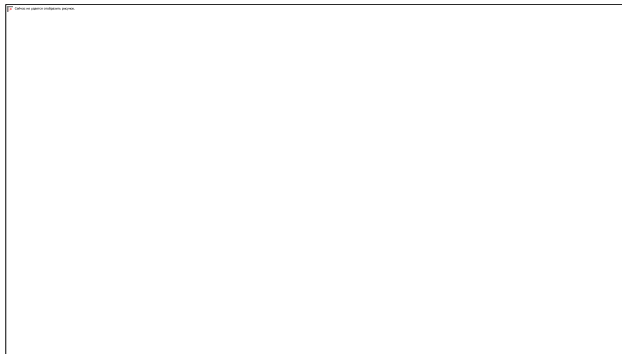


Рисунок 1 – Простейшая конфигурация Mesh-сети на основе модулей ESP32 с применением технологии ESP-Now

Встроенный Wi-Fi модуль позволяет связать роботов между собой в mesh-сеть используя технологию ESP-Now [4]. Таким образом роботы могут обмениваться данными между собой. А множество библиотек, написанных под эту платформу, позволяет сократить время на разработку программного комплекса системы.

Для обнаружения соседей используется система, основанная на инфракрасных передатчиках и приёмниках. Для определения взаимного расположения могут применяться технологии Bluetooth или Wi-Fi. Роботы подключаются в Mesh-сеть, благодаря этому появляется возможность обмениваться данными. Конфигурация простейшей Mesh-сети приведена на рисунке 1. Обе технологии позволяют определять уровень сигнала передатчика. Таким образом, появляется возможность определить расстояние до соседнего робота.

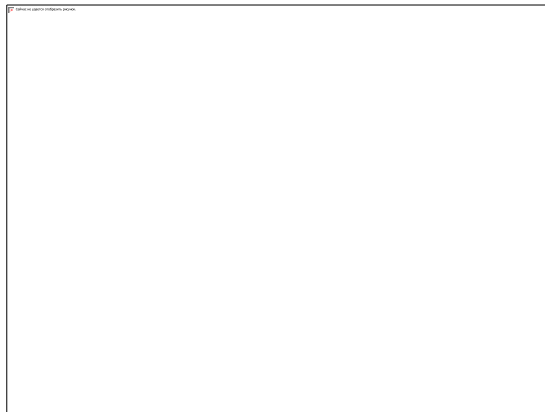


Рисунок 2 – Триангуляция на основании затухания радиосигнала

Зная расстояние между множеством роботов можно вычислить их взаимное расположение путём использования триангуляции радиосигнала, по принципу, схожему с принципом работы системы GPS [5], однако в данном случае расстояние до другого робота (спутника) определяется не на основании разницы во времени, а на основании затухания радиосигнала, диаграмма триангуляционного алгоритма, на основании затухания радиосигнала приведена на рисунке 2.

В случае применения ИК сенсоров [6] также используется триангуляция. Для удешевления системы используется несколько недорогих фотоприёмников расположенных по кругу, таким образом можно определить, с какой стороны находится сосед, переместившись на определённое расстояние и повторно измерив угол взаимного расположения на основании триангуляции можно вычислить расстояние до соседнего робота. Схема расчёта взаимного расположения роботов с применением инфракрасных сенсоров приведена на рисунке 3. При увеличении числа роботов и обмене информацией о взаимном расположении можно добиться высокой точности определения взаимных координат [7].

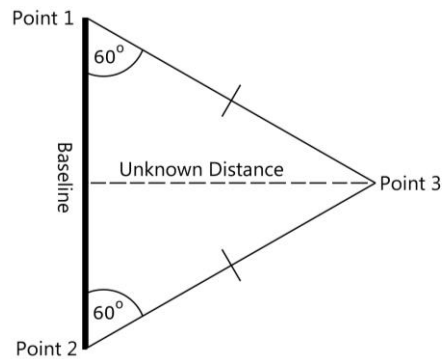


Рисунок 3 – Схема триангуляции с расчётом расстояния на основании углов, измеренных инфракрасными сенсорами

Разрабатываемая робототехническая платформа может применяться для моделирования алгоритмов роевого поведения, для наглядной демонстрации алгоритмов коллективного взаимодействия, что может быть применимо в образовательном процессе для облегчения усвоения материала, а создание копий роботов и доработка платформы студентами позволит развить необходимые инженерные навыки

**Список использованных источников:**

1. Gartner Hype Cycle [Электронный ресурс]. – Режим доступа : <https://www.gartner.com/smarterwithgartner/5-trends-emerge-in-gartner-hype-cycle-for-emerging-technologies-2018/>
2. Карпенко Л. И. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой : учебное пособие / А. П. Карпенко. — Москва : Издательство МГТУ им. Н. Э. Баумана, 2014. —446, [2] с.: ил.
3. Спецификация микроконтроллеров семейства ESP32 [Электронный ресурс]. – Режим доступа : [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf)
4. Инструкция по применению технологии ESP-now [Электронный ресурс]. – Режим доступа : [https://espressif.com/sites/default/files/documentation/esp-now\\_user\\_guide\\_en.pdf](https://espressif.com/sites/default/files/documentation/esp-now_user_guide_en.pdf)
5. Цифровые системы управления технологическим оборудованием : уч. пособие / В. Т. Минченя [и др.]. – Минск : Энциклопедикс, 2016. – 108 с.
6. Пей, А. Сопряжение ПК с внешними устройствами / А. Пей. – М. : ДМК Пресс, 2012. – 320 с.
7. Justin Yonghui Kim. Designing an Effective Millirobot for Swarm Behaviour Studies with Human-Swarm Interaction / Justin Yonghui Kim –Toronto, Mechanical And Industrial Engineering University of Toronto,2017

## **ИДЕНТИФИКАЦИЯ НЕИСПОЛЬЗУЕМЫХ ЗЕМЕЛЬ НА СНИМКАХ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ**

*Леванков А.Н.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Дудкин А.А. – проф. Кафедры ЭВМ, д. т. н, проф.*

В данной работе решается задача идентификации неиспользуемых и залежных (заброшенных) земель на снимках дистанционного зондирования Земли. Объектом исследования являются многоспектральные снимки полученные при помощи космических спутников Sentinel и Landsat. Формируется обучающая выборка, на основе которой создаётся нейросетевой алгоритм идентификации и строится карта неиспользуемых земель.

Залежные земли – это выведенные из севооборота участки бывшей пашни. После прекращения сельскохозяйственной деятельности на них начинает формироваться естественный растительный покров. В отличие от обрабатываемых земель по залежам не ведется подробной статистики. Решение задачи автоматического определения залежных земель может быть особенно важно для территории Республики Беларусь. В период 80-90х годов 20 века большие площади сельскохозяйственных земель республики стали заброшенными и неиспользуемыми. Причиной этому



были авария на Чернобыльской АЭС, в результате чего обширные земли были выведены из оборота в следствии их загрязнения радионуклидами, а также ослабление экономических связей с республиками, раньше входившими в состав СССР. Уменьшение сельского населения, его отток в города, привел к недостатку рабочей силы в сельской местности. В результате всех этих факторов многие земли были заброшены. Со временем, уменьшение радиационного загрязнения и увеличение экономической активности даёт повод задуматься об экономической целесообразности возвращения залежных земель в сельскохозяйственный оборот.

Точное земледелие является одним из инструментов ведения эффективного сельского хозяйства. Его применение предполагает наличие точной и актуальной информации о состоянии растительности и почв. Такую информацию можно получить при помощи геоинформационных технологий и снимков дистанционного зондирования Земли (ДЗЗ). Проблемой является размер данных снимков. Например, у спутника Landsat-8 размер изображений достигает 120 мегабайт для каждого из 11 отдельных спектральных каналов. Встаёт вопрос о том, как эффективно обработать такие большие объёмы данных.

Для анализа состояния растительности широко применяют вегетационные индексы. Это показатель, рассчитываемый в результате операций с разными спектральными диапазонами ДЗЗ, и дающий информацию о параметрах растительности в данном пикселе снимка. Наибольшую информацию о состоянии растительности дают спектральные каналы красного и ближнего инфракрасного диапазонов, так как на красную зону спектра (0,62–0,75 мкм) приходится максимум поглощения солнечной радиации хлорофиллом, а на ближнюю инфракрасную зону (0,75–1,3 мкм) – максимальное отражение энергии клеточной структурой листа [1]. На основе данных этих каналов вычисляют нормализованный вегетационный индекс (NDVI), который чаще всего применяется при исследовании силы вегетации растений.

На данный момент существуют работы позволяющие отделить залежные земли от пашни на снимках ДЗЗ с использованием метода дискриминантного анализа [2]. Сущность метода заключается в расчете классификационных функций, которые вычисляют значения, позволяющие отнести объект к тому или иному классу. Применительно к проблеме дистанционного выделения залежей или пашни такими переменными могут выступать значения NDVI, вычисленные по снимкам поверхности, сделанным в течении всего вегетационного сезона. Точность выявления достигает 70%.

Однако последние достижения в использовании нейросетей (НС) показывают, что их использование может повысить как точность распознавания, так и в большей мере автоматизировать процесс. Как показали предыдущие исследования в распознавании здоровой и слабой растительности, использование значений NDVI для построения на их основе нейросетевой модели может быть эффективно использовано для построения карт сельскохозяйственных полей [3, 4].

Рабочая гипотеза предполагает, что совокупность сезонных значений NDVI для залежных земель отличается от таковых у обрабатываемых земель. В связи с тем, что спектр залежи большую часть вегетационного периода слабо отличается от обычного поля, был выбран подход к получению серии снимков, сделанных на протяжении всего вегетационного периода. Будем исходить из того, что спектральный отклик залежи на протяжении года будет более равномерен и стабилен, чем спектр пахотного поля. Хотя бы раз в год в спектре пашни могут быть обнаружены признаки недавней распашки. Вероятность его обнаружения увеличивается в осенний период, когда сельскохозяйственные земли резко меняют спектральный отклик в следствии распашки или уборки урожая. Данный подход был описан в работе Терёхина Э.А. [2]. Однако вместо статистического анализа и расчёта дискриминантных функций, которые классифицируют поля и залежи, текущая работа предусматривает создание и обучение нейронной сети, которая способна самостоятельно идентифицировать неиспользуемые земли на снимках ДЗЗ.

Цель настоящей работы заключается в оценки возможностей автоматизированного выявления неиспользуемых и залежных земель с помощью нейросетевого алгоритма, обученного на собранном массиве данных, а также демонстрация большей эффективности данного подхода по сравнению с расчётом на основе дискриминантного анализа.

Исходные данные для идентификации – растровые изображения земельных участков, результат идентификации – изображения с выделенными на них неиспользуемыми сегментами.

При этом необходимо решить следующие задачи:

- Снимки разнесённые во времени могут иметь разный угол наклона к меридиану и смещение относительно центра, поэтому требуется предварительная коррекция размеров участка сфотографированной территории и его поворота. Эти действия необходимы для того, чтобы пиксели, соответствующие одним и тем же координатам, совпадали на различных снимках.
- Произвести вычисление нормализованного относительного индекса биомассы NDVI для каждого снимка. Данный этап может быть выполнен с помощью программы геоинформационной обработки данных ArcGIS.
- На основе полученных значений NDVI для каждого пикселя изображений обучающего набора данных сформировать вектор, состоящий из сезонных значений NDVI.

- Определить архитектуру нейронной сети и обучить её на наборе сформированных пар <вектор сезонных значений NDVI, идентификатор класса>, где идентификатор класса принимает единичное значение, если пиксель принадлежит искомому сегменту, иначе - нулевое.

Обученная НС далее используется для попиксельного распознавания заброшенных полей. Точность может быть повышена при использовании ансамбля НС.

Архитектура нейросетевой модели представляет собой перцептрон с одним скрытым слоем и нелинейной функцией активации гиперболический тангенс. Обучение производится по методу обратного распространения ошибки. Размер входного слоя НС определяется размерностью вектора сезонных значений NDVI. Размерность скрытого слоя выбирается эмпирически. Размер выходного слоя равен 2, где одно значение соответствует классу залежных земель, а второе всем остальным. Исходными данными для алгоритма является обучающая выборка с изображениями полей размером порядка 1000 элементов.

В результате исследования были сделаны следующие выводы:

- Алгоритм предусматривает сбор и использование данных о состоянии вегетации полей на протяжении одного вегетационного сезона. Необходим анализ спектральных характеристик полей на разных этапах вегетационного периода с целью обнаружения спектра почвы, который отличает распаханную землю от неиспользуемых.
- Использование нейросетевой модели позволяет повысить точность дистанционных методов мониторинга сельскохозяйственных полей по сравнению с методами, основанными на использовании дискриминантного анализа, а также добиться большей автоматизации процесса.

**Список использованных источников:**

1. Чаюк А.И., Клебанович Н.В. Оценка динамики состояния растительности с использованием вегетационных индексов
2. Терехин, Э.А. Распознавание залежных земель на основе сезонных значений вегетационного индекса NDVI / Э.А. Терехин // Компьютерная оптика. – 2017. – Т. 41, № 5. – С. 719-725.
3. Ганченко В.В., Дудкин А.А., Марушко Е.Е. Построение карт сельскохозяйственных полей по аэрофотоснимкам различного спектрального диапазона для систем точного земледелия
4. Ганченко В.В., Дудкин А.А. Распознавание состояния сельскохозяйственной растительности по данным аэрофотосъемки на основе сверточных сетей

## **МЕТОДИКА УСТРАНЕНИЯ ШУМА НА ОКТ-ИЗОБРАЖЕНИЯХ**

*Лисовский А.Г.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Одинец Д.Н. – кандидат технических наук*

На основе результатов экспериментального исследования методов удаления спекл-шумов на изображениях, полученных при оптической когерентной томографии, разработана методика обработки ОКТ-изображений, направленная на повышение качества изображений. Для устранения шума использованы гауссовы и медианный фильтры размытия и их комбинации.

Метод оптической когерентной томографии (ОКТ) впервые был предложен в 1991 г., в настоящее время является одним из наиболее перспективных и развивающихся методов неинвазивной диагностики оптически неоднородных сред. ОКТ – это современная технология бесконтактного исследования внутренней микроструктуры объектов с высоким разрешением на глубине проникновения оптического излучения. При проведении ОКТ-исследования детектирование слоев ткани на различных глубинах основано на времени распространения волн излучателя до соответствующего слоя ткани и обратно до приемника.

Разнообразие свойств биологических тканей определяет изменения их линейных размеров в результате прикладываемого к объекту внешнего воздействия. Это ведет к различному изменению концентрации рассеивателей в биологической ткани. Для ОКТ диагностики характерно наличие спекл-шума, который можно наблюдать в виде пятен, появляющихся на структурных изображениях.

Спекл-шум носит мультипликативный характер. Данный тип шума является неотъемлемым атрибутом на изображениях полученных в ходе исследований методом оптической когерентной томографии, из-за которого разрешение и контраст изображений уменьшаются, что влияет на диагностическую ценность этого метода визуализации. Появление спекл-шума обусловлено когерентностью волн в плечах интерферометра и хаотически распределенными отражателями излучения. Такой шум, как правило, труднее удалить, т.к. интенсивность шума зависит от интенсивности изображения. Модель мультипликативного шума определяется формулой:

$$y_{ij}=X_{ij}n_i(1),$$

где  $y_{ij}$  – спекл-изображение,  $X_{ij}$  – исходное изображение,  $n_{ij}$  – негауссовский шум. Индексы  $i, j$  представляют пространственное положение на изображении.

Уменьшение спекл шума улучшает результат визуального и цифрового распознавания объектов на изображении. При использовании методов машинного обучения в анализе ОКТ данных, предварительно обработанные изображения позволяют базовой модели достичь высокой точности по сравнению с более сложной моделью, обученной на изображениях, которые не были предварительно обработаны. Подавление спекл-шума является важным этапом предварительной обработки и должно проводиться, не затрагивая важных характеристик изображения.

Для устранения шума на изображении и выделения контуров объектов применяются цифровые фильтры. В процессе фильтрации значение яркости в каждой точке исходного изображения заменяется значением, которое считается наименее искаженным помехой. Применение цифрового фильтра предполагает получение свертки изображения с определенным ядром. В работе рассматриваются гауссов и медианный фильтры, являющиеся основой комбинаций, используемых для устранения шума.

Гауссов фильтр используется для размытия изображения с целью удаления шума. При применении данного фильтра интенсивность пикселей центрального элемента в пределах ядра заменяется средневзвешенным значением окружающих пикселей. Для однозначного определения центрального элемента, размер окна должен быть нечётным. Матрица перемещается по изображению, при этом весовая функция в процессе перемещения остается неизменной.

Еще одним типом фильтров для устранения шума является медианный фильтр. Медианная фильтрация основана на замене значений отсчетов в центре окна медианным значением исходных отсчетов внутри апертуры фильтра, т.е. центральный элемент окна заменяется медианой всех элементов изображения внутри апертуры.

Применение фильтров размытия приводит к потере информации о контурах объектов на изображении. Для повышения четкости изображений в работе использовался метод выравнивания гистограмм, заключающемся в распределении яркости элементов изображения на основе анализа статистического распределения элементов изображения с различной яркостью. Гистограммы очень темных изображений характеризуются тем, что ненулевые значения гистограммы сконцентрированы около нулевых уровней яркости, а для очень светлых изображений наоборот – все ненулевые значения сконцентрированы в правой части гистограммы. Наиболее удобным для восприятия человеком будет изображение, у которого гистограмма близка к равномерному распределению. Для улучшения визуального качества к изображению надо применить такое преобразование, чтобы гистограмма результата содержала все возможные значения яркости и при этом в примерно одинаковом количестве. В результате выравнивания гистограммы в большинстве случаев существенно расширяется динамический диапазон изображения, что позволяет отобразить ранее не замеченные детали.

В ходе эксперимента были рассмотренные алгоритмы устранения шума были применены к ОКТ-снимкам. Исходный вид изображения и этого же изображения после устранения спекл-шума гаусс фильтром с ядром 5x5 представлены на рисунках 1 и 2 соответственно.

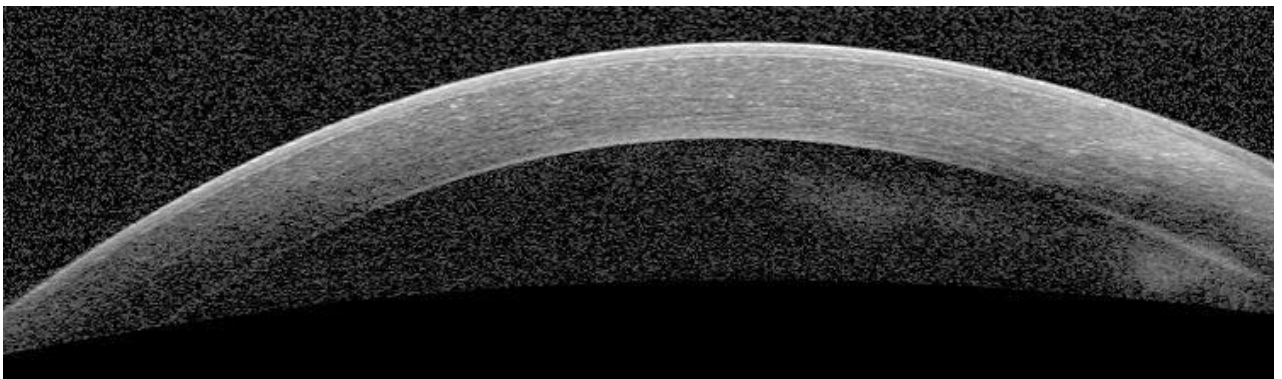


Рисунок 1 – Исходное ОКТ-изображение

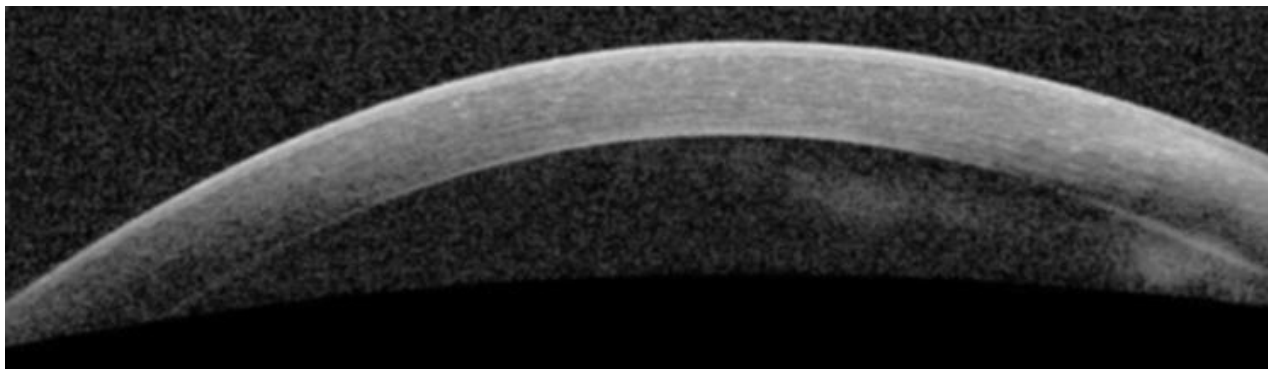


Рисунок 2 – Изображение после устранения спекл-шума гаусс фильтром с ядром 5x5

Для оценки качества обработки изображений были использованы следующие критерии: количество несовпадающих пикселей и соотношение сигнал/шум. Первый критерий отображает степень сходства между исходным и обработанным изображением, второй сравнивает уровень полезного сигнала с уровнем фонового шума. Помимо качественной оценки результатов применения фильтров оценивается время вычислений фильтра. Время вычислений ( $T_c$ ) фильтра определяется как время, затрачиваемое цифровой вычислительной платформой для выполнения алгоритма фильтрации, когда на нем не запущено никакое другое программное обеспечение, кроме операционной системы.

Рассмотренные методы устранения спекл-шума на практике были применены к набору ОКТ-снимков сетчатки глаза, вычислены значения приведённых метрик качественной оценки изображений и зафиксировано время выполнения программного модуля. Полученные результаты отражают зависимость качества обработки изображений от параметров фильтра и количества итераций их применения. Повысить качество позволяет применение комбинаций фильтров, в качестве одного из элементов которых выступает гаусс фильтр. Отмечен прирост времени необходимого на обработку изображения при применении нескольких фильтров к изображению, что необходимо учитывать при одновременной обработке большого количества изображений.

**Список использованных источников:**

1. P.S. Hiremath, Prema T. Akkasaligar, Sharan Badiger. Speckle Noise Reduction in Medical Ultrasound Images [Электронный ресурс]. – Режим доступа: <https://www.intechopen.com/books/advancements-and-breakthroughs-in-ultrasound-imaging/speckle-noise-reduction-in-medical-ultrasound-images/>. – Дата доступа: 10.04.2020.
2. Галёб К.И.С., Проскурин С.Г. УМЕНЬШЕНИЕ СПЕКЛ-ШУМОВ ПРИ ПОСТРОЕНИИ СТРУКТУРНОГО ОКТ ИЗОБРАЖЕНИЯ // Фундаментальные исследования. – 2014. – № 4-3. – С. 479-483
3. Бобкова А.О., Поршнева С.В., Зюзин В.В., Бобков В.В. Исследование методов удаления спекл-шумов на ультразвуковых изображениях [Электронный ресурс]. – Режим доступа: <https://www.graphicon.ru/html/2013/papers/244-246.pdf> – Дата доступа: 10.04.2020.

## **ПРЕИМУЩЕСТВА ЭЛЕКТРОННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ФОРМИРОВАНИЕМ РЕЧЕВЫХ НАВЫКОВ ВЛАДЕНИЯ ИНОСТРАННЫМ ЯЗЫКОМ**

*Марзалюк А.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Дадыкин А.К. – канд. техн. наук, доцент*

В настоящее время необходимость коммуникации на иностранном языке обусловлена в связи с глобализацией и развитием информационных технологий. Существующие приложения и сервисы изучения иностранного языка не удовлетворяют задаче быстрого освоения навыков владения иностранным языком. Решением этой задачи служит применение Системы Управления Формированием Речевых Навыков (СУФН) с Структурно-Визуальным Методом (СВМ), в которой делается упор на развитие речевых навыков (слушания и говорения) с привязкой к профессиональным навыкам. Применение СУФН позволит существенно повысить качество обучения и сократить время на формирование навыков владения иностранным языком до 6 месяцев.

В настоящее время необходимость коммуникации на иностранном языке обусловлена в связи с глобализацией и развитием информационных технологий. Английский язык господствует в мире как язык науки, межнационального общения, язык IT-технологий. Необходимость повсеместного обучения английскому языку не ослабевает, а с течением времени все усиливается. При всём распространении курсов и изучения его в средней школе и ВУЗе, уровень владения английским языком остаётся

низким у большинства его изучавших [1]. Также с трудом решается задача формирования навыка беглого общения, что сильно затрудняет коммуникацию. В свете недавних событий с миграционным кризисом задача быстрого формирования навыков владения иностранным языком приобрела большую значимость.

Основными речевыми навыками являются: слушание, говорение, чтение, письмо. Слушание и говорение, являясь навыками устной речи, играют главенствующую роль в повседневной речевой деятельности. Основной проблемой традиционного подхода к обучению иностранным языкам с использованием грамматико-переводного метода является его упор на развитие письменной речи в ущерб устной. Он широко используется в курсах, самоучителях, системе образования. Применение данного подхода приводит к появлению эффекта «внутреннего перевода», когда в сознании обучающегося предложение строится на родном языке, а затем переводится на иностранный и наоборот, что затрудняет использование языка и способствует появлению ошибок. Во избежание этого эффекта предлагается применять структурно-визуальный метод (СВМ) [2].

В деле обучения иностранному языку широкое распространение получили электронные системы управления обучением. В отличие от традиционных методов обучения они обладают рядом преимуществ:

1. Гибкость – одно из самых главных преимуществ. Позволяет учитывать индивидуальные возможности учащегося, самостоятельно выбирать подходящий темп, время и место проведения занятий. Повышается доступность обучения для людей с ограниченными возможностями.

2. Экономическая выгода – нет необходимости в оснащении и содержании учебных помещений. Учебные материалы доступны удаленно в электронном виде, при необходимости они легко обновляются.

3. Последовательность – вне зависимости от квалификации и мастерства преподавателя, образовательный процесс носит субъективный характер. Одна и та же тема различными преподавателями будет представлена абсолютно по-разному. Эти факторы могут пагубно воздействовать на последовательность, которая должна быть присуща любому образовательному курсу. Электронное обучение всегда последовательно, так как обучающийся может оставаться на одном этапе так долго, пока не достигнет желаемого результата.

4. Ответственность за образовательные результаты – электронное обучение требует, чтобы обучающиеся были самомотивированными на обучение. Система позволяет учащемуся чувствовать себя ответственным за свое образование. Это чувство ответственности в сочетании с чувством независимости создают ситуацию успеха, что способствует достижению образовательных целей.

Система Управлением Формированием Речевых Навыков владения иностранным языком (СУФН) является развитием таких систем. Основной особенностью СУФН является применение СВМ и речевых технологий, что обеспечивает формирование языковых (слушание и говорение) и профессиональных навыков в сознании обучающегося [3]. Она включает в себя:

Систему управления контентом (СУК);  
интерактивные речевые тренажеры (ИРТ), работающих с одновременным использованием визуальной лексики и визуальных моделей (ВМ) обретаемого языка;  
систему распознавания и моделирования речи (СМР);  
система непрерывной оценки (СНО) текущей языковой компетенции обучающихся в сочетании с виртуальным помощником (ВП).

Каждый компонент системы вносит свой вклад в деле формирования навыков владения языком. В процессе работы системы прогресс обучающегося непрерывно оценивается СНО, давая данные для корректировки процесса обучения. ВП на основе результатов СНО и ранее подготовленных уроков с помощью СУК формирует уроки для ИРТ. В процессе работы ИРТ используется СМР для голосового взаимодействия с обучающимся, что позволяет заговорить с первых минут.

В процессе работы системы проводится подробный статистический анализ результатов, отображаются динамические кривые обучения каждого взрослого обучающегося, обосновываются таблицы коэффициентов и уточняются показатели скорости формирования навыков речи [4], определяются уровни говорения от начального до уровня спонтанного говорения в соответствии со шкалой Общеввропейских компетенций владения иностранным языком (CEFR) [5]. Обобщенная структура СУФН представлена на рисунке 1.

Преимуществами СУФН являются:

Обеспечивается формирование речевой зоны обретаемого языка в сознании обучающегося за счет резкого повышения эффективности упражнений с помощью трансформации грамматической информации из словесной в графическую форму;

использование достижений ИКТ-сферы в качестве инструмента обеспечения реализации целей обучения с непрерывным мониторингом текущего состояния и получением гарантированного результата обучения за конечное число шагов;

непрерывная оценка уровня компетентности обучающегося приводит к формированию

логарифмической зависимости кривой обучения и компенсирует предпосылки ее деградации, обеспечивая возникновение синергетического эффекта в процессе обучения;

использование СУФН возможно с компьютеров и мобильных телефонов с возможностью выхода в сеть интернет. Нет необходимости в учебных помещениях и в прямом взаимодействии с преподавателем;

формирования процесса обучения на основе сознания обучающегося, а не программы обучения.

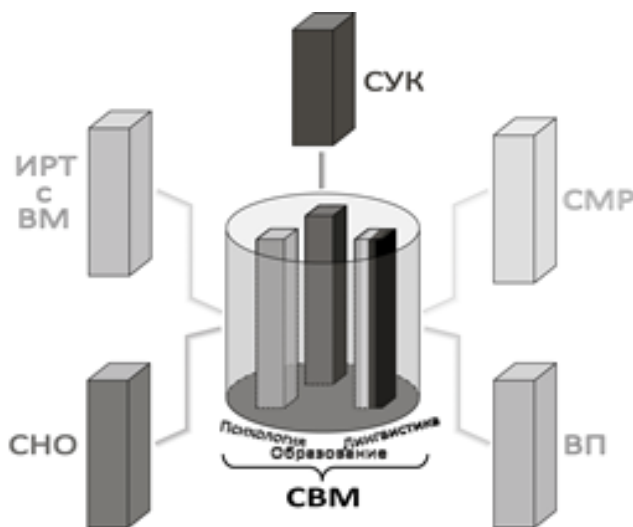


Рисунок 1 – Обобщенная структура СУФН

СУФН – это уникальный наукоемкий высокотехнологичный продукт. Её использование приведет к синергетическому эффекту в процессе обучения и, как следствие, ускоренному созданию новой языковой зоны в сознании взрослых обучающихся. Система обеспечивает процесс управляемого формирования профессиональных и речевых навыков, что позволяет перейти от изучения языка к его совершенствованию в процессе использования.

**Список использованных источников:**

1. EF Education First [Electronic resource]. EF English Proficiency Index 2019. – 2019. – Mode of access: <https://www.ef.com/~media/centralegcom/epi/downloads/full-reports/v9/ef-epi-2019-english.pdf>. – Date of access: 28.03.2020
2. Dadykin, A.K. The Visual Approach in Educational Projects / A.K. Dadykin, V.A. Dibrova, I.H. Tahini // International Journal of Social Science and Humanity. – 2017. – Vol. 7, No. 6. – P. 373–377.
- 3 А.К. Дадыкин, А.В. Марзалюк. Концепция системы управления формированием языковых и профессиональных навыков взрослых на основе информационно-коммуникационных технологий / X Международная научно-техническая конференция «Информационные технологии в промышленности, логистике и социальной сфере» (IT'2019): тезисы докладов, Минск, 23-24 мая 2019 г. – Минск: ОИПИ НАН Беларуси, 2019. – 270 с. – ИСБН: 978-985-7198-01-06. – с. 65-67.
4. Варгас, Дж. Анализ деятельности учащихся. Методология повышения школьной успеваемости / Дж. Варгас.– М. : Оперант, 2015. – 480 с.
5. Council of Europe Language Policy Portal [Electronic resource]. Common European Framework of Reference for Languages: Learning, teaching, assessment (CEFR). – 2018. – Mode of access: <https://rm.coe.int/cefr-companion-volume-with-new-descriptors-2018/1680787989>. – Date of access: 23.03.2020. – P. 173–198.

## АРХИТЕКТУРА РАСПРЕДЕЛЕННОЙ СИСТЕМЫ ОБРАБОТКИ ВИДЕО

Пац Е.А.

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Насуро Е.В. – к.т.н., доцент каф. ЭВМ*

Архитектура распределенной системы обработки видео на основе микросервисов и облачных технологий. Основными преимуществами системы являются: высокая доступность, автоматическая масштабируемость, отказоустойчивость.

В современном мире большую популярность получают облачные технологии. Применение облачных технологий значительно снижают затраты на инфраструктуру, уменьшают время доставки кода и увеличивают доступность сервисов для конечных пользователей. Существуют различные виды предоставления облачных услуг. Каждый уровень имеет свою зону ответственности и модель обслуживания как для провайдера, так и для конечного пользователя. Ключевыми критериями являются время доступности и SLA.

К основным видам предоставления услуг относятся: программное обеспечение как сервис (Software as a service, SaaS), платформа как сервис (Platform as a service, PaaS), инфраструктура как сервис (Infrastructure as a service, IaaS). Все данные уровни разграничивают ответственность между провайдером облачных услуг и заказчиком. Разграничение ответственности подробно изображено на рисунке 1.

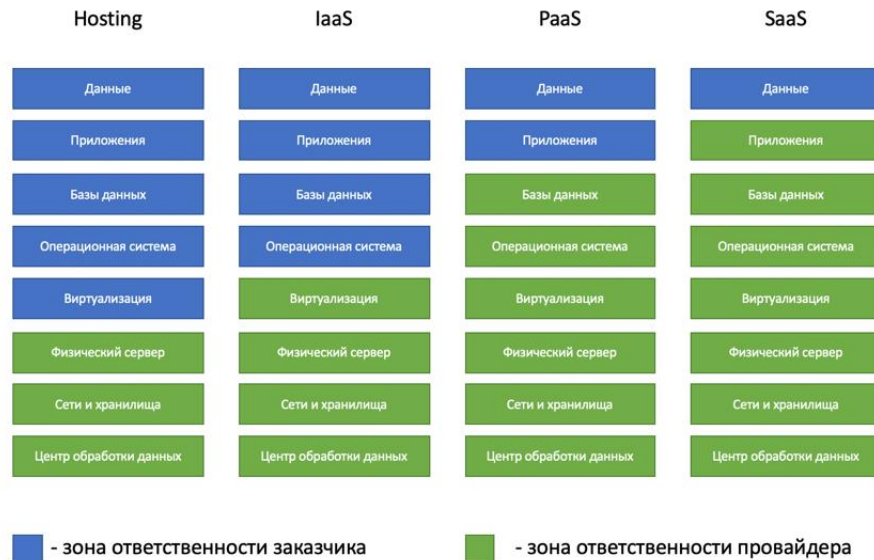


Рисунок 1 – Принцип разграничения ответственности облачных услуг

Для построения распределенной системы достаточно использовать модель предоставления IaaS, так как на данном уровне облачные провайдеры берут на себя ответственность за виртуализацию. Это означает, что в случае отказа физического сервера виртуальные машины запущенные на нем будут мигрированы на исправные сервера с минимальным временен недоступности. Таким образом система не зависит от аппаратного обеспечения и все инциденты решаются на стороне провайдера. Так же, при таком подходе, отсутствует зависимость от определенного облачного провайдера, таких как Google Cloud Platform (GCP), Amazon Web Services (AWS), Microsoft Azure, Alibaba Cloud. Существует вариант использования приватного облака на основе OpenStack. Появляется возможность изменить облачного провайдера без изменения программного кода и архитектуры платформы.

Микросервисная архитектура подразумевает собой развертывание приложений в контейнерах. Существует множество платформ, осуществляющих управление и оркестрацию. К таким платформам относятся OpenStack (RedHat), Kubernetes (Google), Mesos (Apache), Docker Swarm. Для решения поставленной задачи идеально подходит платформа Kubernetes. Данная платформа разработана на языке программирования, что позволяет его модифицировать. Так же огромное количество крупных проектов использует данную платформу. Kubernetes позволяет с легкостью масштабировать кластер используя API облачного провайдера, что позволяет экономить на ресурсах. Особую роль в платформе играет механизм репликации, это повышает доступность конечных сервисов и позволяет распределять нагрузку. К преимуществам Kubernetes можно отнести то, что данная платформа не диктует выбор определенного типа приложений или фреймворка, кроме этого, для разработки приложений можно использовать любой язык программирования. При необходимости развертывания баз данных, возможно использование как SQL (PostgreSQL, MySQL, Oracle), так и NoSQL (MongoDB, Cassandra). Kubernetes предоставляет абстракции для кластеризации и репликации баз данных, возможность и методы бекапирования. Использование данной платформы позволяет проводить обновления сервисов и приложений без времени недоступности.

Основная функция Kubernetes заключается в оркестрации контейнеров что понимается, как планирование работы контейнеров разной степени загруженности на физических и виртуальных устройствах. Так же платформа контролирует ограничения, налагаемые средой развертывания и конфигурацией кластера. Kubernetes имеет механизмы слежения за всеми запущенными контейнерами, по средствам которых проводится анализ состояния приложений.

Использование контейнеров произвело настоящий прорыв в парадигме разработки и выполнения крупных, сложных программных систем. По сравнению с традиционными моделями они имеют такие преимуществ. Во-первых, использование микросервисной архитектуры уменьшает время разработки приложений, так же уменьшается время их развертывания, что позволяет прибегнуть к

методологиям непрерывной разработки, интеграции и развертывания. Разграничивается ответственность между разработчиком и администратором, так как поставка производится в виде образов контейнеров. Образ представляет собой шаблон из набора слоев, используемый для запуска контейнера.

Запуск контейнеров происходит в определенной среде выполнения контейнеров, что позволяет с легкостью воссоздавать окружения для разработки, тестирования и промышленного использования. Так же, отсутствует привязка к аппаратному обеспечению и операционной системе вычислительной машины, на которой будет запущен контейнер. Принцип работы среды запуска контейнеров подробно изображен на рисунке 2.



Рисунок 2 – Принцип работы среды запуска контейнеров

В качестве среды выполнения контейнеров был выбран Docker. На текущий момент данное решение является самым востребованным. Docker имеет множество функций, которые отсутствуют у конкурентов. Так же имеется площадка DockerHub на которой собрано большое количество образов позволяющих решать различные задачи. Для построения системы приложение необходимо упаковать в контейнер, предварительно создав образ. Для разработки образа необходимо использовать Dockerfile.

Таким образом, разработанная архитектура соответствует основным параметрам, каждый компонент системы отказоустойчив и имеет возможность горизонтального масштабирования. В разработанной архитектуре применены самые актуальные облачные решения.

**Список использованных источников:**

1. Sean P. Kane, Karl M. Docker: Up & Running: Shipping Reliable Containers in Production – O'Reilly Media, 2015 – 201 с.
2. Ричардсон К. Микросервисы. Паттерны разработки и рефакторинга – Санкт-Петербург : Издательский дом «Питер», 2020. – 544 с.
3. Сайфан Джиджи Осваиваем Kubernetes. Оркестрация контейнерных архитектур. – СПб.: Питер, 2019. – 400 с.
4. Ньюмен С. Создание микросервисов – Санкт-Петербург : Издательский дом «Питер», 2016. – 304 с.

## ДИНАМИЧЕСКАЯ СИМУЛЯЦИЯ ОБЪЕМНОГО ОГНЯ

*Стаховский А.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Кукин Д.П. – канд. техн. наук*

При выборе алгоритма для динамической симуляции необходимо найти баланс между реалистичностью симуляции и скоростью симуляции. Для успешного выполнения данной задачи ключевое значение имеют хорошее понимание структуры симуляции, знание преимуществ и недостатков популярных алгоритмов, используемых в различных подзадачах симуляции.

Симуляция трехмерного огня в режиме реального времени находит свое применение в различных интерактивных приложениях. Среди интерактивных приложений, анимации огня наиболее востребованы в видеоиграх. В видеоиграх необходимость симуляции огня была с самого момента их появления, однако, всего два десятилетия назад стало возможным использовать огонь в трехмерных сценах. В компьютерной графике довольно часто требуется найти компромисс между скоростью и



реализмом. В приложениях реального времени, скорости отрисовки отдается наибольший приоритет; увеличенный реализм бесполезен, если частота кадров не дотягивает до определенного уровня. Поэтому основной проблемой рендеринга в реальном времени является поиск таких алгоритмов, которые позволяют получить достаточную реалистичность, при которой частота кадров будет не менее минимального порога приемлемой частоты кадров[1].

В общем случае задача симуляции огня может быть разбита на три непересекающихся подзадачи[2]:

- моделирование;
- анимация;
- визуализация.

В первую очередь необходимо выбрать подходящую внутреннюю структуру, или модель, для симуляции. Далее, требуется выбрать способ анимации — метод, с помощью которого будет происходить взаимодействие с моделью. Техника анимации служит для того, чтобы оживить модель, привести ее в движение. Наконец, модель и ее анимацию необходимо отрисовать на экране, используя для этого некоторые примитивы визуализации (полигоны, текстуры, сферы, воксели и т.п.).

Альтернативная схема была предложена в [3], в которой авторы делают акцент на алгоритмах распространения огня. Фаза моделирования в данном случае является одним из этапов при разработке алгоритмов распространения пламени.

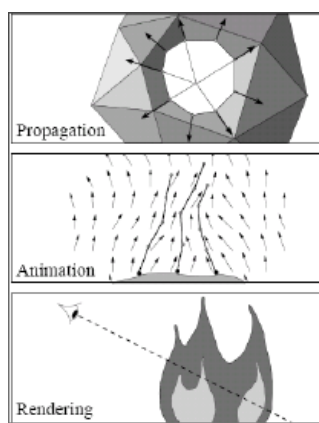


Рисунок 1 – Основные задачи симуляции, согласно [2]

**Список использованных источников:**

1. firestarter – A Real-Time Fire Simulator. / M. de Kruijf, M. Schneider, 2007.
2. Synthesizing Flames and their Spreading / C. H. Perry, R. W. Picard // Proceedings of the Fifth Eurographics Workshop on Animation and Simulation, 1994. – P. 1–14.
3. Realistic and Controllable Fire Simulation / P. Beaudoin [et al.] // Graphics Interface, 2001. – P. 159-166.

## ПОДХОД К СОЕДИНЕНИЮ МНОЖЕСТВА КАМЕР С RASPBERRY PI ДЛЯ СЪЕМКИ СФЕРИЧЕСКИХ ПАНОРАМ

*Стракович А.И.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Луцик Ю.А. – к.т.н, доцент*

Показан подход к получению множества изображений для создания сферической панорамы, основанный на использовании множества камер. Рассмотрены проблемы которые могут возникнуть при подключении камер и способы их решения.

Сферическая панорама представляет из себя собранное из множества отдельных кадров изображение в сферической проекции. Характерной чертой сферических панорам является максимально возможный угол обзора пространства (360×180 градусов). Просматривая такие панорамы создается эффект присутствия на рассматриваемом месте.

Для создания сферических панорам, необходимо получить множество фотографий одновременно либо за минимальное время, чтобы движущиеся объекты находились на одном и том же месте на разных фото. Иначе совместить такие фото будет невозможно, либо на итоговой панораме случайно попавший в кадр движущийся объект будет прозрачным.

Основным управляющим устройством, для работы с камерами, может выступать практически

любой микроконтроллер с GPIO. В представленном подходе используется плата Raspberry PI 3B+[1]. Используемые камеры OV2640[2]. Для работы с камерой необходим тактирующий сигнал с частотой 24 МГц или больше. Он используется для синхронизации выдачи камерой очередного байта изображения и принятия данного байта Raspberry. Механизм синхронизации можно реализовать несколькими способами.

Первый подход – генерация сигнала самой Raspberry. Программной генерацией, в зависимости от используемых библиотек, можно получить частоту от 15 кГц до 22 МГц[3]. Недостатком данного подхода является то, что он медленный и нестабильный, с точки зрения получаемой частоты.

Второй подход – использование внешнего тактового генератора. Такой генератор всегда будет выдавать стабильный сигнал необходимой частоты. Необходимо только согласовать напряжение генератора и Raspberry, если генератор формирует сигнал отличный от требуемого для Raspberry PI.

У Raspberry PI 28 GPIO пинов. Для подключения камеры нужны 14 пинов. Из них 12 для получения изображения и 2 для инициализации камеры по интерфейсу SCCB (совместим с I2C). В таком случае получится подключить только 2 камеры. Возникает проблема нехватки пинов для подключения всех камер одновременно. В качестве согласующего элемента предусмотрен блок коммутации, который в определенный момент времени передает данные изображения только одной из камер. Реализацию блока коммутации можно выполнить несколькими способами.

Первый способ заключается в реализации отдельной схемы, что представляется аппаратно более избыточным.

Второй способ предполагает использование ПЛИС, на которой реализовать схему из первого способа. Данный подход является более гибким, т.к. при любом изменении конфигурации коммутирующего устройства не требуется изготавливать новую схему, а только загрузить конфигурацию в ПЛИС. Для реализации этого подхода используется ПЛИС EPM240T100C5[4].

Общая схема представлена на рисунке 1. К выбранной ПЛИС можно подключить 6 камер. При использовании камер с объективом типа “рыбий глаз” их хватит для полного покрытия обзора 360x180 градусов, что позволит сделать сферическую панораму.

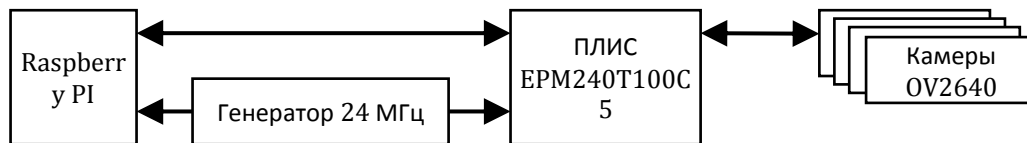


Рисунок 1 – Схема подключения

**Список использованных источников:**

1. Raspberry PI 3B+ Datasheet [Электронный ресурс] – Режим доступа : [www.raspberrypi.org/documentation/hardware/computemodule/datasheets/rpi\\_DATA\\_CM3plus\\_1p0.pdf](http://www.raspberrypi.org/documentation/hardware/computemodule/datasheets/rpi_DATA_CM3plus_1p0.pdf) – Дата доступа : 01.03.2020.
2. Камера OV2640 Datasheet [Электронный ресурс] – Режим доступа : [www.uctronics.com/download/cam\\_module/OV2640DS.pdf](http://www.uctronics.com/download/cam_module/OV2640DS.pdf) – Дата доступа : 01.03.2020.
3. Benchmarking Raspberry Pi GPIO Speed [Электронный ресурс] – Режим доступа : [www.codeandlife.com/2012/07/03/benchmarking-raspberry-pi-gpio-speed/](http://www.codeandlife.com/2012/07/03/benchmarking-raspberry-pi-gpio-speed/) – Дата доступа : 10.03.2020.
4. EPM240T100C5 Datasheet [Электронный ресурс] – Режим доступа : [www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/max2/max2\\_mii5v1.pdf](http://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/max2/max2_mii5v1.pdf) – Дата доступа : 12.03.2020

## МОНИТОРИНГ И ПРОГНОЗИРОВАНИЕ УСТАЛОСТИ ВОДИТЕЛЯ ЗА РУЛЕМ

Фоменок В.В., Шакун Р.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Лукашевич М.М. – к.т.н., доцент

Для классификации усталости или состояния тревоги водителя использовались искусственные нейронные сети. В последние десятилетия искусственные нейронные сети широко используются для классификации изображений. Чтобы проанализировать состояние сонливости водителя, было получено 200 изображений водителя во время обычного процесса вождения. Сто изображений содержат изображения с открытыми глазами или полуоткрытыми глазами, а еще сто изображений содержат изображения с закрытыми глазами.

### Искусственная нейронная сеть со скрытым слоем

После получения 200 изображений 140 из них были использованы для обучения, проверки и тестирования нейронной сети: 70 с открытыми глазами или полуоткрытыми глазами и 70 с закрытыми глазами. Остальные изображения (30 для открытых или полуоткрытых глаз и 30 для закрытых глаз)

оставляли для тестирования сети после завершения тренировочного процесса. Сеть обучалась с использованием структуры, представленной на рисунке 1 (2601 нейрон во входном слое, 10 нейронов в скрытом слое и 2 нейрона в выходном слое). Количество нейронов во входном слое соответствует количеству элементов входного вектора, который является версией с пониженной дискретизацией изображения драйвера с преобразованием в один столбец (представленной матрицей уровня серого 51x51 элементов). Количество нейронов в выходном слое соответствует числу возможных категорий, по которым можно классифицировать изображение (2 категории: усталость или тревога).

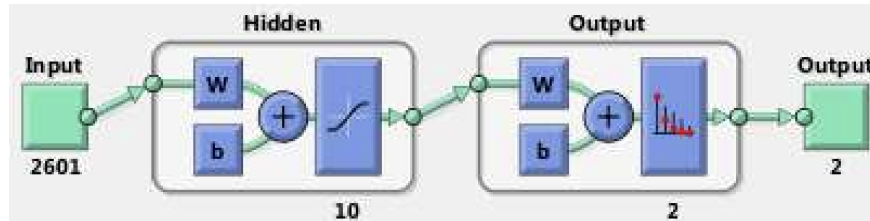


Рис. 1. Нейронная сеть с одним скрытым слоем

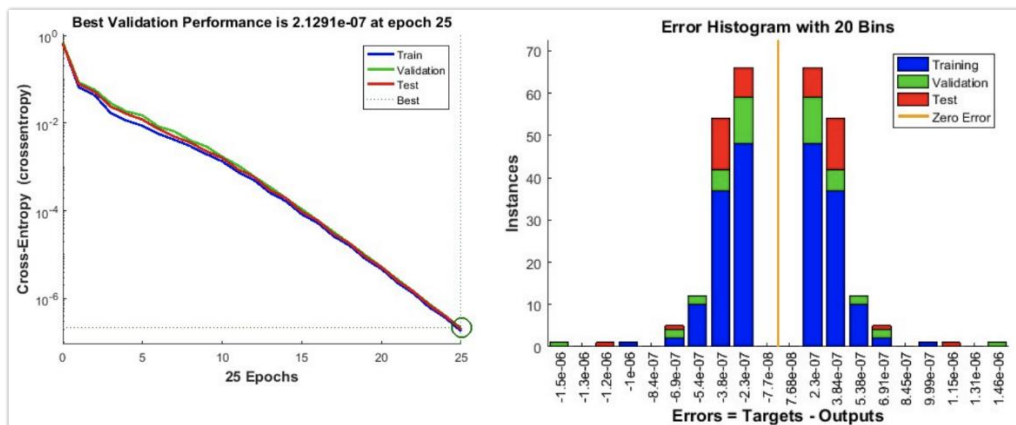


Рис. 2. Диаграмма производительности и гистограмма ошибок обученной сети.

В процессе обучения были получены диаграмма результатов обучения и гистограмма ошибок обученной сети, которые представлены на рисунке 2. Можно заметить, что тренировочные показатели достигают значений менее  $10^{-6}$  в 25 эпохах, а границы гистограммы находятся в диапазоне от  $-1,5 \times 10^{-6}$  до  $1,46 \times 10^{-6}$ , что является очень хорошими результатами.

### Искусственная нейронная сеть с автоматическим кодированием

Для сети с автоматическим кодированием использовались те же входные данные, что и в случае сети со скрытым слоем. Автоэнкодеры используют методы для отдельного обучения каждого слоя, затем объединяют их в одну сеть с несколькими слоями и обучают конечную сеть в целом. Структура сети представлена на рисунке 3.

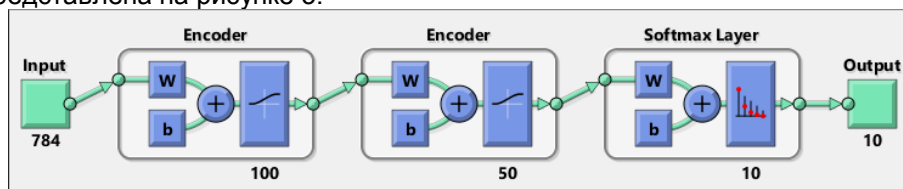


Рис. 3. Нейронная сеть с двумя скрытыми слоями автоэнкодеров

Результаты обучения автоэнкодера представлены на рисунке 4. На рисунке слева представлена производительность тренировки, когда она достигла значения менее 0,034 после 438 эпох. На рисунке справа результаты теста показывают, что не было получено ложных срабатываний или ложных отрицательных результатов. Это означает, что каждое тестовое изображение (60 изображений - 30 для каждого класса) было правильно классифицировано.

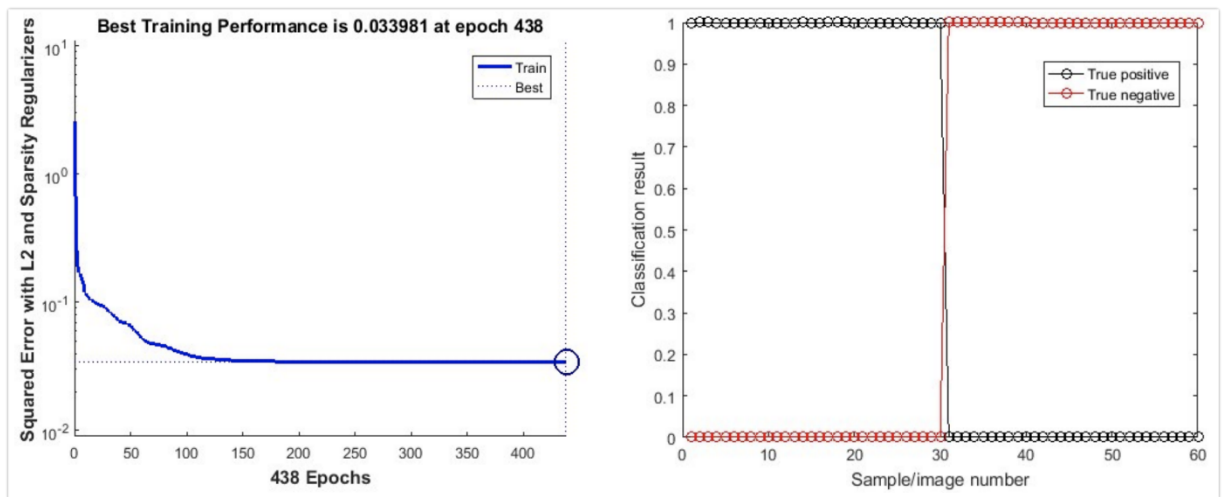


Рис. 4. Производительность обучения и результаты тестирования для сети автоэнкодеров.

### Заключение

Анализируя результаты применения нейронных сетей к полученным изображениям, можно сделать вывод, что обе сети дали очень хорошие результаты со 100% положительными результатами классификации. Небольшое количество нейронов, используемых в скрытых слоях для успешной классификации изображения (10 для сети с 1 скрытым слоем и 15 для сети с автоматическим кодированием) позволяют реализовать эти сети на компактных вычислительных устройствах, используя очень небольшую часть их памяти. Кроме того, время обработки составляет порядка миллисекунд. Обучение сети может быть сделано специально для каждого водителя, таким образом увеличивая вероятность успеха классификации.

#### Список использованных источников:

1. Волчек А.А., Пойта П.С., Шведовский П.В. Мониторинг, оценка и прогноз чрезвычайных ситуаций и их последствий – Брест: Альтернатива, 2012. – 426 с.
2. Riedmiller, M. Advanced supervised learning in multi-layer perceptrons—from backpropagation to adaptive learning algorithms //Computer Standards & Interfaces. – 1994. – V. 16. – №. 3. – P. 265-278.
3. Морковкина Е. А. Исследование профессионально-важных качеств водителей транспортных средств // Молодой ученый. 2009. №3. С. 143-151.

## ВОССТАНОВЛЕНИЕ ИЗОБРАЖЕНИЙ С ПОМОЩЬЮ НЕОБУЧЕННЫХ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ

Хурсов П. С.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Искра Н. А. – ст. преп.

Зачастую возникает необходимость в восстановлении искаженных изображений. Примерами искажений могут быть артефакты сжатия, наложение шума или посторонних объектов. В данной работе предлагается использование необученных сверточных нейронных сетей для решения этой задачи.

Типичным решением для задачи восстановления изображения является обучение какой-либо генеративной модели на обучающей выборке и использование её для генерации восстановленного изображения. Однако недавние исследования[1] показали, что обучение не является необходимостью, так как некоторое знание о выборке может содержаться прямо в структуре модели. Так сверточным нейронным сетям гораздо проще работать с сигналами, нежели чем с шумом. Это можно объяснить тем, что основным слоем в сверточных нейронных сетях является сверточный слой, который можно рассматривать как некоторый цифровой фильтр над выходом предыдущего слоя.

Используя это можно сформулировать задачу восстановления как генерацию изображения из некоего случайного шума:

$$\operatorname{argmin}_{\theta} \|(D(x) - x_0)\|, x = f_{\theta}(z^*) \quad (1),$$

где  $D(x)$  – некий деградационный процесс над изображением  $x$ ,  $x_0$  – исходное изображение,

$f_{\theta}(z^*)$  - процесс генерации изображения моделью с параметрами  $\theta$  из шума  $z^*$ .

Таким образом производя минимизацию целевой функции, можно добиться того, чтобы модель могла генерировать целевое изображение из случайного шума.

В качестве модели используется U-Net[2](Рис. 1) с некоторыми изменениями в конфигурации. Модель состоит из 5 блоков понижения\повышения размерности. Блок понижения размерности состоит из сверточного слоя с шагом 2 и 128 фильтрами, слоем групповой нормализации, еще одного сверточного слоя из 128 фильтров, групповой нормализации и функции активации LeakyReLU. Блок повышения размерности имеет похожую конфигурацию, но вместо сверточного слоя с шагом 2 используется билинейная интерполяция для повышения размерности. Для минимизации целевой функции(1) использовался алгоритм Adam[3].

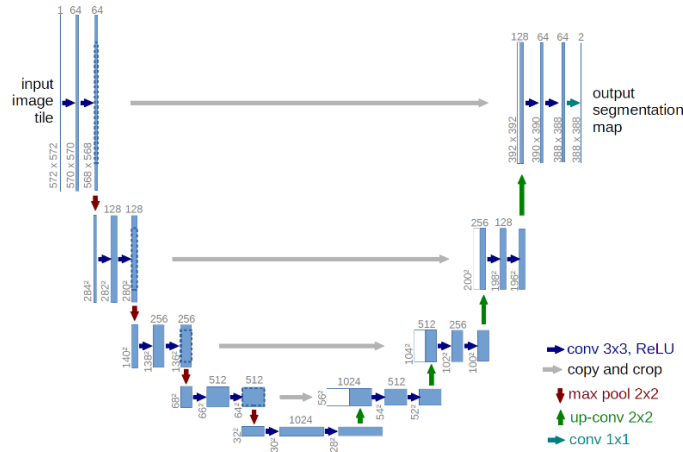


Рисунок 1 – Архитектура нейросети U-Net.

Так как алгоритму понадобится некоторое количество шагов минимизации прежде чем начать воспроизводить помимо сигнала шум, то для решения задачи удаления шума можно использовать экспоненциальное скользящее среднее над результатами каждого шага алгоритма минимизации.

$$I_{n+1} = I_n * \beta + I * (1 - \beta) \quad (2),$$

Где  $I_{n+1}$  – значение скользящего экспоненциального среднего на  $n+1$  итерации,  $I_n$  – значение на  $n$  итерации,  $\beta$  – взвешивающий параметр.

Результат удаления шума можно увидеть на рисунке 2.



Рисунок 2 – Результат восстановления зашумленного изображения. Слева направо: оригинальное изображение, изображение с шумом, восстановленное изображение.

Уровень искажения зашумленного изображения по метрике Peak Signal to Noise Ratio(PSNR)

$$PSNR(I, K) = 10 \log_{10} \left( \frac{255}{\frac{1}{NM} \sum_{i,j} (I_{i,j} - K_{i,j})^2} \right) \quad (3),$$

Где  $I, K$  – пара изображений размера  $N$  на  $M$  пикселей.

Как видно из формулы, чем больше изображения похожи друг на друга, тем большим будет метрика PSNR. Для пары оригинального и зашумленного изображения PSNR равен 26.02, а для пары оригинального и восстановленного 30.38. Как видно необученные нейронные сети вполне могут быть использованы для удаления шума с изображений.

Другим применением необученных нейронных сетей для восстановления изображений может

быть задача супер разрешения. Задача состоит в том, чтобы по изображению низкого разрешения получить изображение большего разрешения без потери качества. Для этого нужно в выражении 1 использовать в качестве деградационного процесса функцию понижения разрешения сгенерированного изображения до оригинального.

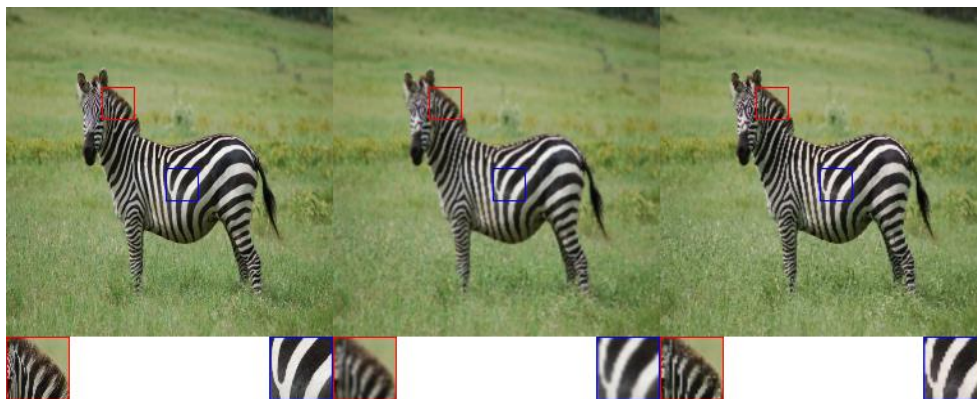


Рисунок 3 – Результат повышения разрешения изображения. Слева направо: оригинальное изображение, изображение восстановленное с помощью бикубической интерполяции, изображение восстановленное с помощью необученной нейронной сети.

Для сравнения использовалось изображение в высоком разрешении (512x512 пикселей). Далее используя метод бикубической интерполяции разрешение понижалось до 256x256 пикселей. После этого разрешение снова повышалось до 512x512 пикселей с помощью бикубической интерполяции и с помощью необученной сверточной нейронной сети (Рис. 3).

Если сравнить изображения по метрике PSNR, то пара оригинального изображения и восстановленного с помощью бикубической интерполяции имеет значение PSNR 26.16, а пара оригинального и восстановленного с помощью необученной нейронной сети 28.22.

Как видно необученные нейронные сети могут быть эффективным инструментом для решения задач восстановления изображений. Однако для этого необходимо знать деградационный процесс. Так, например, можно восстанавливать размытые изображения, если известно ядро размытия.

**Список использованных источников:**

1. Deep Image Prior / D. Ulyanov, A. Vedaldi, and V. Lempitsky // CVPR, 2018.
2. U-net: Convolutional networks for biomedical image segmentation / O. Ronneberger, P. Fischer, and T. Brox . – MICCAI, 234–241c, 2015.
3. Adam: A method for stochastic optimization / Diederik P. Kingma and Jimmy Lei Ba // ICLR, 2015.

## ИССЛЕДОВАНИЕ ПОДХОДОВ К РЕАЛИЗАЦИИ ОПЕРАЦИИ СВЁРТКИ НА ОСНОВЕ ТЕХНОЛОГИИ CUDA

*Хурсов П. С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Перцев Д. Ю. – ст. преп.*

Одной из наиболее распространенных операций над изображениями является фильтрация на основе операции свертки. Для решения данной задачи используются графические ускорители с поддержкой технологии CUDA. В данной работе приводится обзор различных подходов к реализации операции свертки.

Наиболее распространенным форматом изображений является RGB888, при котором каждый пиксель изображения кодируется 3 байтами, описывающие интенсивность красной, зеленой и синей компоненты. При таком формате изображение зачастую хранится в памяти в виде плоского массива размерностью  $N \times M \times 3$  байта, где  $N$  – высота,  $M$  – ширина изображения.

Для эффективной работы с памятью в CUDA необходимо следовать некоторым правилам, в частности, формировать транзакции при работе с глобальной памятью. Для этого необходимо, чтобы адреса памяти, к которым обращается waгр, были выравнены на границу 128 байт. При работе с одномерным массивом пикселей данное условие легко нарушить, так как адрес начала некоторой строки зависит от ширины изображения. Чтобы исправить данный недостаток используется:

– функция `cudaMallocPitch`, которая выделяет расширенный объем памяти для матрицы  $N \times M$ , добавляя некоторое количество памяти в конец каждой строки, чтобы сделать адрес начала каждой

строки гарантированно кратным 128;

– cudaMemcpy2D, которая позволяет эффективно работать с выравненной памятью.

Альтернативным подходом является использование текстурной памяти [1]. Однако данный тип памяти может работать только с 1-, 2- и 4-байтными словами. Для работы с текстурной памятью необходимо ввести альфа-канал, что увеличит расход памяти и потребует некоторой предобработки изображения. Сравнение скорости работы операции свертки над выравненной и не выравненной матрицами приведен на рис. 1. Так как основным узким местом на данном этапе является обращение к исходному изображению, сравнение ведется по размеру изображения при фиксированной ширине, что наглядно демонстрирует важность выравнивания начала строк в памяти.

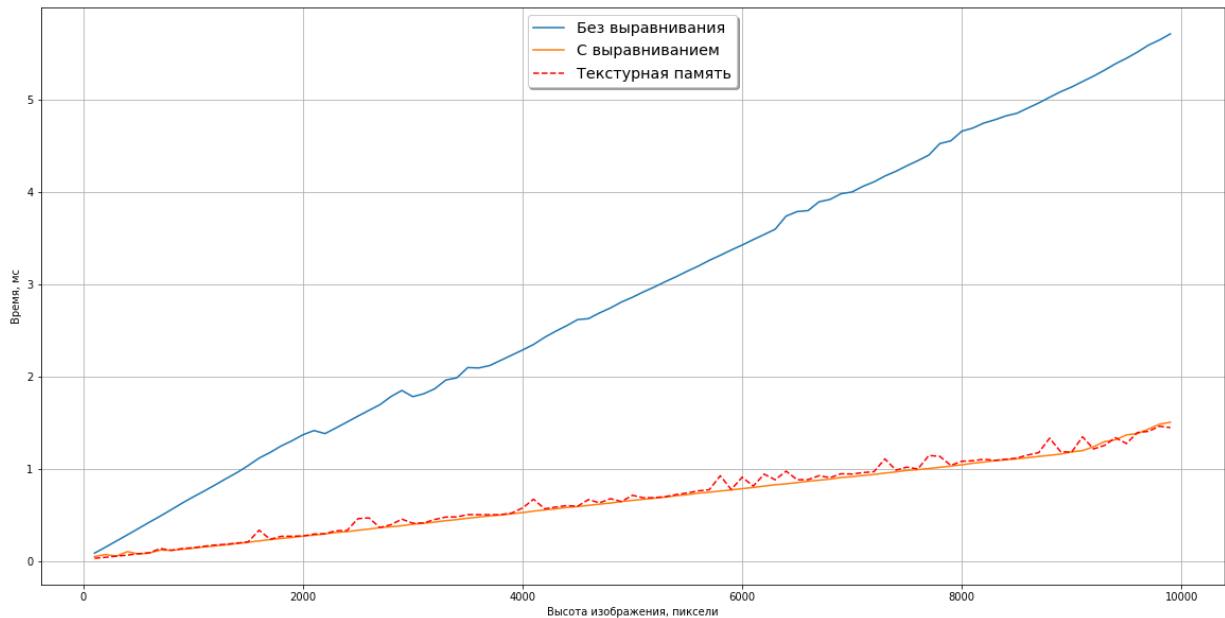


Рисунок 1 – Сравнения времени обработки изображения различными способами работы с памятью

Как видно, наилучшими способами работой с памятью являются выравненная и текстурная память. Также видно, что по времени они не сильно отличаются и между ними следует выбирать исходя из других факторов, например, объема памяти. Также NVIDIA API обязывает использовать выравненную память для инициализации текстурной, однако текстурная память предоставляет более удобный способ работы с границами изображения.

Другим условием формирования 128-байтной транзакции является обращение к 4-байтным словам. Однако, при использовании глобальной памяти каждый пиксель задается 3 байтами, т. е. при подходе одна нить – один пиксель формирования транзакции не удастся достичь. Если посмотреть на расположение изображения в памяти (рис. 2), видно, что при подходе одна нить – четыре пикселя возможно достигнуть формирования трех 128-байтных транзакций.



Рисунок 2 – Расположение изображения в памяти.

Свертка задается выражением:

$$D[i, j] = \sum_{m, n} S[i - m, j - n] * K[m, n] \quad (1)$$

где  $D$  – матрица отфильтрованного изображения,  $S$  – матрица исходного изображения,  $K$  – ядро свертки,  $i, j$  – координаты пикселя.

При вычислении соседних пикселей значения из матрицы исходного изображения и ядра свертки используются по несколько раз. Технология CUDA предоставляет в распоряжение

программисту разделяемую память. Доступ к разделяемой памяти для потоков в пределах одного блока исполняемого ядра CUDA гораздо быстрее, чем доступ к глобальной или текстурной памяти. Используя это можно существенно ускорить выполнение операции свертки.

Время выполнения также зависит от типа арифметики. Основными вариантами являются целочисленная и арифметика с одинарной точностью (арифметика с двойной точностью в данной задаче имеет мало смысла, так как формат изображения целочисленный и нет необходимости в повышенной точности). Сравнение различных вариантов реализации ядра с использованием разделяемой памяти и без, а также различных типов данных показан на рис. 3.

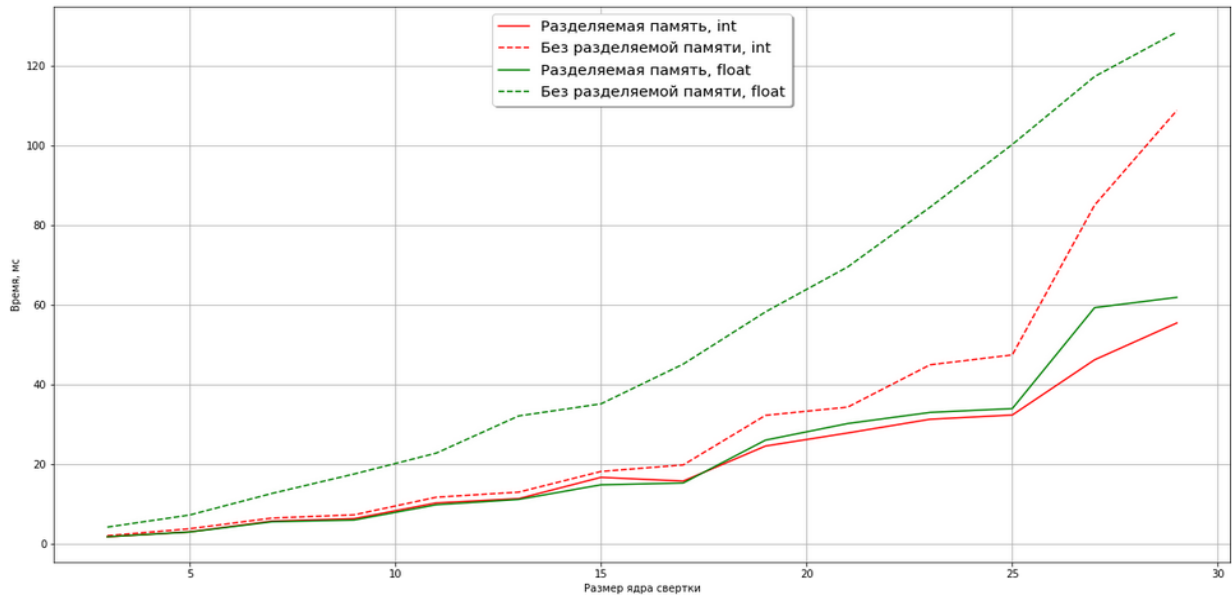


Рисунок 3 – Зависимость времени выполнения операции свертки от размера ядра матрицы свертки.

Как видно из графиков оптимальным с точки зрения скорости исполнения является использование разделяемой памяти с целочисленной арифметикой. Таким образом, оптимальной с точки зрения использования памяти и быстродействия является реализация, использующая выравненную глобальную память, разделяемую память внутри блока и целочисленную арифметику.

**Список использованных источников:**

1. Основы работы с технологией CUDA /Боресков А. В., Харламов А. А. // М.: ДМК Пресс, 2010. – С. 129-131.

## ШУМОПОДАВЛЕНИЕ НА ИЗОБРАЖЕНИИ С ПОМОЩЬЮ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ

Шакун Р.А., Фоменок В.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Лукашевич М.М. – к.т.н., доцент

FFDNet - это новейший метод шумоподавления изображений, основанный на архитектуре сверточной нейронной сети. В отличие от других существующих нейросетей, FFDNet имеет несколько преимуществ, таких как быстрое время выполнения и использование небольшого объема памяти, а также способность обрабатывать широкий диапазон уровней шума с единой сетевой моделью. Сочетание качества шумоподавления и небольшой вычислительной нагрузки делает этот алгоритм привлекательным для практического использования в шумоподавляющих приложениях. В этой работе будут обсуждаться характеристики архитектуры этого алгоритма.

В последнее время, благодаря выдающимся характеристикам, привлекли значительное внимание новые способы шумоподавления на изображении, основанные на методах глубокого обучения. В частности, дискриминационные методы обучения (например CSF, TNRD) показывают относительно быстрое время вывода и очень хорошие шумоподавляющие характеристики. Тем не менее, существенным недостатком всех этих алгоритмов является то, что конкретная модель должна быть обучена для каждого уровня шума.

Другой популярный подход включает использование сверточных нейронных сетей (CNN), таких, как RBDN, DnCNN и FFDNet. Их производительность выгодно отличается от других



современных алгоритмов как количественно, так и визуально. Эти методы состоят из последовательности сверточных слоев с нелинейными активационными функциями между ними. Наиболее интересной из перечисленных сверточных сетей является FFDNet [1].

Для реализации FFDNet используется стандартная архитектура прямой связи, как показано на рисунке 1. Сеть состоит из  $D$  сверточных слоев, которые имеют одинаковую структуру. Каждый из них имеет  $W$  выходов, а пространственный размер их ядер  $K \times K$ .

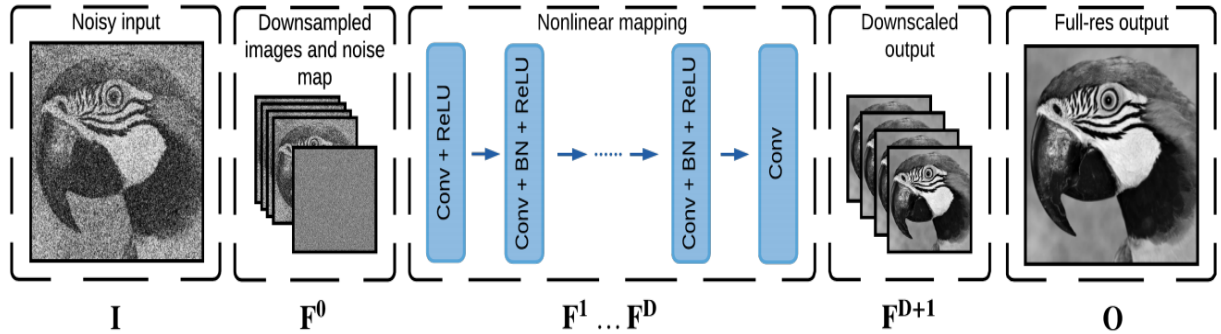


Рисунок 1 – Архитектура FFDNet.

На уровне предварительной обработки сеть сначала реорганизует пиксели  $I$  входного изображения размером  $n_{ch} \times h \times w$  в изображение с более низким разрешением размером  $4n_{ch} \times h/2 \times w/2$ .

Слой  $F^0$  извлекает  $2 \times 2$  патчи и реорганизует их пиксели в разных каналах выходного изображения в соответствии с  $F^0$ :

$$F^0[c, x, y] = I \left[ \left\lfloor \frac{c}{4} \right\rfloor, 2x + (c \bmod 2), 2y + \left\lfloor \frac{c}{2} \right\rfloor \right] \quad (1),$$

где  $0 \leq c < 4n_{ch}$ ,  $0 \leq x < h$ ,  $0 \leq y < w$ . Большая часть обработки будет выполнена в этом уменьшенном масштабе.

Также ко входу добавляется карта оценки шума  $M$  такого же разрешения. Эта шумовая карта контролирует компромисс между шумоподавлением и сохранением детализации. Для пространственно инвариантного гауссовского шума со стандартным отклонением  $\sigma$ , карта шума однородна и все ее элементы равны  $\sigma$ . На рисунке 2 показана схема этого слоя.

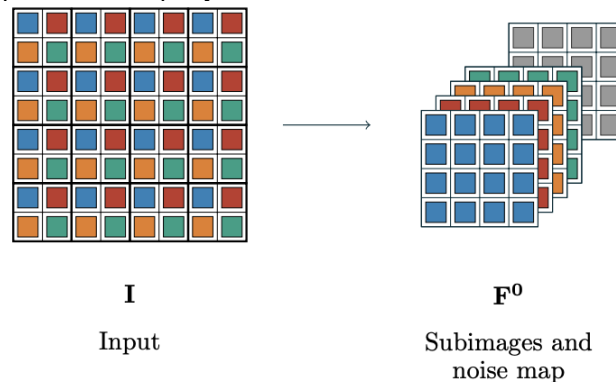


Рисунок 2 – Схема слоя масштабирования.

Нелинейное отображение, состоящее из  $D$  сверточных слоев ( $F^1 \dots F^D$ ) следует после предварительной обработки слоя  $F^0$  (рисунок 1). Каждый из этих слоев состоит из  $W$  сверточных фильтров пространственного размера  $K \times K$ . Выходы слоев  $F^1 \dots F^{D-1}$  следуют за точечной функцией активации  $ReLU(\cdot) = \max(\cdot, 0)$ . Во время обучения слои нормализации помещаются между сверточным и  $ReLU$  слоями от  $F^2$  до  $F^{D-1}$ . Во время оценки слои нормализации удаляются и заменяются аффинным слоем.  $c$ -й канал  $d$ -го слоя,  $F^{dc}$  можно записать как:

$$F_c^d = ReLU \left( \sum_{c'=0}^{D-1} w_{cc'}^d * F_{c'}^{d-1} \right) \text{ for } c \in \{0 \dots W - 1\} \quad (2),$$

где  $w_{cc'}^d$  - двумерное ядро свертки размера  $K \times K$ . Обобщая характеристики из  $D$  слоев



Семантическая сегментация изображений заключается в выделении на изображении локальных областей (сегментов), соответствующих различным классам объектов. Ручная сегментация - длительный и трудозатратный процесс. Автоматизация данного процесса является актуальной задачей хотя бы с экономической точки зрения. На сегодняшний день существует множество методов и алгоритмов классификации объектов на изображениях, в том числе на снимках земной поверхности, обеспечивающих хорошие результаты работы. Однако данным алгоритмам присущи определенные недостатки.

В отличие от задач локализации и детектирования объектов на изображении, задача семантической сегментации является более сложной. Это обусловлено необходимостью не только верно определить принадлежность объекта на изображении к определенному классу, но и максимально четко определить границы и структуру объекта на изображении. Наиболее распространенным на сегодняшний день методом семантической сегментации изображений в задачах точного земледелия является применение свёрточных нейронных сетей. Например, в работе [1] производится сегментация и инжирных плантаций. Предложенный в статье метод основан на свёрточной нейронной сети SegNet [2]. Средняя точность составила 93.85%.

В статье [3] предлагается алгоритм сегментации корней и почвы, основанный на U-Net [4]. Кроме того, авторы провели сравнение с Frangi-Net [5]. Обе архитектуры показывают сходные результаты (точность 99.7% у U-Net против 99.6% у Frangi-Net).

В работе [6] представлено описание аппаратно-программной платформы для сегментации с целью нахождения сорняков. В основе алгоритма сегментации лежит свёрточная нейронная сеть, содержащая 3 свёрточных слоя и 2 полносвязных слоя. Заявленная точность составила 97%.

В работе [7] так же решается подобная задача. В данном случае используется комбинация DenseNets [8] и FCN [9]. Авторы заявляют о точности в 98%.

Как видно, использование свёрточных нейронных сетей в решениях задач сегментации позволяет добиваться хороших результатов. Так, в приведённых примерах точность составляла не менее 90%. Исходные данные

#### Исходные данные

Материалами для исследования являлись фотографии полей картофеля, произведённые в разное время суток и с различной высоты. Особенностью изображений растительности является то, что цвет растений зависит от погодных условий, времени суток и времени года в момент съёмки. Кроме того, различные болезни также влияют на окраску растений. Стоит учитывать, что данные факторы могут уменьшать точность сегментации.

Для обучения нейронной сети была произведена разметка исходных изображений. Для каждого исходного изображения была сделана маска, где белым цветом обозначался участок с картофелем. Набор данных для обучения содержал в себе 260 изображений.

Представленная работа фокусируется на сегментировании участков с картофелем. Классификация растения и оценка его состояния не проводилась.



Рисунок 1 – Пример фотографии картофельного поля и его разметки

Особенностью метода, предложенного в данной работе, является то, что в качестве функции потерь используется Dice Loss, который был представлен в работе [10], а в качестве метода оптимизации – Adam [11]. Входом нейронной сети является тензор, размерностью 512x512x3, выходом – тензор 512x512x1. В качестве метрики для оценки качества работы алгоритма была выбрана точность (precision). Точность при обучении составила 98.3%. Точность при тестировании – 97.2%.

#### Анализ полученных результатов

В качестве работы для сравнения выбрана [12]. Авторами предлагается алгоритм разделения изображений на 2 класса (почва и растение). В качестве основы алгоритма используется свёрточная нейронная сеть, имеющая 2 свёрточных слоя и 1 полносвязный слой. Размер входного слоя 33x33x3. Обучающая выборка содержала 110528 изображений (20% использовались для валидации).

Результирующая точность на валидационной выборке составила 96.9%.

Как видно, предложенный алгоритм на основе U-Net показывает лучшие результаты (98.3% против 96.9%) в задаче двухклассовой классификации. Более того, размер обучающей выборки, несопоставимо меньше (260 изображений против 110528).

Предложенный в настоящей работе алгоритм способен обрабатывать 10 изображений в секунду на ноутбуке с графическим ускорителем NVIDIA GeForce GTX 1060 with Max-Q Design, центральным процессором Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz и 16GB оперативной памяти под управлением ОС Ubuntu 16.04.

Вывод

Была реализована система технического зрения на основе фреймворка PyTorch с применением нейронной сети с архитектурой U-Net. Система позволяет производить сегментацию фотографий с целью нахождения участков с сельскохозяйственной растительностью (картофелем). Особенностью является выбор функции Dice Loss в качестве функции потерь и метод Adam в качестве оптимизатора при обучении нейронной сети. При тестировании была достигнута точность 97.2%.

#### Список использованных источников:

1. Fig Plant Segmentation from Aerial Images Using a Deep Convolutional Encoder-Decoder Network [Электронный ресурс]. – Режим доступа: [https://www.researchgate.net/publication/333131201\\_Fig\\_Plant\\_Segmentation\\_from\\_Aerial\\_Images\\_Using\\_a\\_Deep\\_Convolutional\\_Encoder-Decoder\\_Network](https://www.researchgate.net/publication/333131201_Fig_Plant_Segmentation_from_Aerial_Images_Using_a_Deep_Convolutional_Encoder-Decoder_Network)
2. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation [Электронный ресурс]. – Режим доступа: <https://ieeexplore.ieee.org/document/7803544/>
3. Segmentation of roots in soil with U-Net [Электронный ресурс]. – Режим доступа: <https://plantmethods.biomedcentral.com/articles/10.1186/s13007-020-0563-0>
4. U-Net: Convolutional Networks for Biomedical Image Segmentation [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1505.04597>
5. Frangi-Net: A Neural Network Approach to Vessel Segmentation [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1711.03345>
6. Real-time blob-wise sugar beets vs weeds classification for monitoring fields using convolutional neural networks [Электронный ресурс]. – Режим доступа: <https://www.ipb.uni-bonn.de/pdfs/milioto17uavg.pdf>
7. Fully Convolutional Networks with Sequential Information for Robust Crop and Weed Detection in Precision Farming [Электронный ресурс]. – Режим доступа: <http://www.ipb.uni-bonn.de/wp-content/papercite-data/pdf/lottes2018ral.pdf>
8. The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation [Электронный ресурс]. – Режим доступа: <https://arxiv.org/pdf/1611.09326.pdf>
9. Fully Convolutional Networks for Semantic Segmentation [Электронный ресурс]. – Режим доступа: [https://people.eecs.berkeley.edu/~jonlong/long\\_shelhamer\\_fcn.pdf](https://people.eecs.berkeley.edu/~jonlong/long_shelhamer_fcn.pdf)
10. V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation [Электронный ресурс]. – Режим доступа: <http://far.in.tum.de/pub/milletari2016Vnet/milletari2016Vnet.pdf>
11. Adam — latest trends in deep learning optimization [Электронный ресурс]. – Режим доступа: <https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimization-6be9a291375c>
12. Ганченко В.В., Дудкин А.А. Распознавание состояния сельскохозяйственной растительности по данным аэрофотосъемки на основе сверточных сетей.

## СИСТЕМА НАВИГАЦИИ МОБИЛЬНОГО РОБОТА

*Нагараев М.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Татур М.М. д-р техн. наук, профессор*

В последнее время все больший интерес разработчиков стали привлекать автономные мобильные роботы, функционирующие в повседневном окружении человека – в индустриальной сфере или в сфере обслуживания. Мировой объем таких бытовых роботов уже более 1 миллиона штук.

По причине возрастающих с каждым годом требований к автономности мобильных робототехнических средств существует необходимость создания сложного навигационного комплекса для наземных мобильных роботов, позволяющего расширить их функциональные возможности по ориентации на местности, составлению карт, проходу по безопасным маршрутам и т.п.

Большинство разработок бытовых роботов испытывают дефицит алгоритмов и программ для решения наиболее трудной проблемы – автоматического управления траекторией для достижения цели в помещениях с присутствием большого числа заранее известных помех движению робота.

В данной работе предложен проект системы локальной навигации – одной из основных составляющих такого навигационного комплекса.

Говоря о навигации автономных мобильных роботов, выделяют два основных типа систем – глобальной навигации и ЛН. В данной работе речь пойдет о системах локальной навигации.

Системы ЛН планируют и контролируют выполнение маневров, составляющих движение по

сформированному маршруту. ЛН – схема навигации автономных мобильных устройств, отвечающая за определение относительных координат объекта преимущественно в течение короткого промежутка времени.

Основные задачи ЛН:

- 1) определение относительных декартовых координат и путевого угла мобильного робота, а также его скорости;
- 2) планирование и контроль выполнения роботом маневров, составляющих движение по сформированному маршруту;
- 3) корректировка схем локальной навигации для определения координат аппарата.

В связи с этим выдвигаются следующие требования при разработке таких систем:

- 1) возможность определения углов ориентации мобильного робота;
- 2) непрерывная выдача навигационной информации;
- 3) высокая точность вычисления координат робота в течение коротких промежутков времени;
- 4) автономность работы системы;
- 5) малые габариты и энергопотребление;
- 6) устойчивость к помехам и вибрациям.

Для обнаружения и объезда препятствий используются пара датчиков, сенсоров расстояния, которые позволяют в радиусе 1-1,5 метров определять препятствие. По мере своего продвижения робот выстраивает двумерную карту окружающего пространства. По ней, собственно, прокладывается путь и планируется движение робота вычислителем автопилота.

Данные с датчиков и других сенсоров отправляются с подвижной платформы по Wi-Fi на компьютер. В свою очередь программа отслеживает передвижение робота и, основываясь на полученных данных, составляет карту пространства и базу с координатами.

Система управления мобильным роботом решает 2 ключевые задачи:

- 1) обработка сенсорных данных с целью сбора информации о роботе и внешней среде вокруг него;
- 2) формирование траекторий движения МР, которые бы приводили к выполнению роботом локальной подзадачи;

Результаты испытаний физической модели системы подтверждают возможность ее использования в составе навигационного комплекса наземного мобильного робота. Обеспечиваемая системой точность определения местоположения объекта, а также углов его ориентации позволяет применять разработанную систему как автономно для контроля маневров, составляющих движения объекта по безопасному маршруту, в течение коротких промежутков времени, так и в комплексе с другими навигационными средствами.

**СЕКЦИЯ**  
**«ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ**  
**ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»**

# МЕТОД ОБНАРУЖЕНИЯ И РАСПОЗНАВАНИЯ ДВИЖУЩИХСЯ ОБЪЕКТОВ В ВИДЕОПОТОКЕ

*Альховский А.А., Деменковец Д.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Серебряная Л.В., доцент, к.т.н.*

В работе представлен метод решения задачи обнаружения и распознавания движения объектов для обеспечения требуемого уровня безопасности на промышленных производствах. Используются глубокая искусственная нейронная сеть, одноплатный компьютер, алгоритмы обнаружения движения и сопровождения объектов.

Производственный процесс, в котором участвует человек, несет в себе опасность получения травм. Соблюдения одних лишь требований безопасности недостаточно. Во всем мире ежегодно регистрируется примерно 340 млн. несчастных случаев на производствах. Своевременное обнаружение нестандартных ситуаций может помочь спасти жизнь человека.

В настоящее время является актуальным автоматизированный контроль процесса производства и объектов, участвующих в нем, включая людей. Были определены нормальные режимы производственного процесса, характеризующие безопасное состояние человека в нем.

В работе представлен метод обнаружения движения объектов в видеопотоке с применением одноплатного компьютера Jetson NANO [1] и искусственных нейронных сетей (ИНС). Для достижения поставленной цели необходимо решить следующие задачи: определение состояния объекта и направление его движения. Анализ существующих способов показал, что решение сформулированной задачи на основе датчиков движения, установленных в касках или на оборудовании, достаточно дорогое. Поэтому было решено использовать модель на основе ИНС, данные, получаемые с видеокamеры, установленной на одноплатном компьютере Jetson NANO [1]. Jetson NANO позволяет работать в режиме, близком к режиму реального времени, и имеет графическую мощность, равную 128 ядер, достаточной для работы с глубокой ИНС.

Решение задачи начинается с определения состояния объекта, т.е. движется он или нет. В случае движения его необходимо распознать в видеопотоке, а также уметь отличить от других объектов, присутствующих в видеопотоке. Затем требуется определить, изменялось ли местоположение объекта в видеопотоке, полученном со статической камеры. В качестве обучающей выборки для ИНС использован набор данных под названием COCO (Common Objects in Context) [2]. Набор данных COCO содержит ~64000 размеченных изображений с людьми и отличается простым форматом данных. Для обнаружения объекта в кадре был использован алгоритм YOLO (You Only Look Once) [3]. Этот алгоритм требует получение из исходного кадра признаков изображений. Преобразование исходного кадра в многомерную матрицу признаков выполняется на основе архитектуры ИНС MobileNet v2 [4]. Результат работы алгоритма YOLO представляет собой многомерную матрицу предсказаний ИНС, которая носит избыточный характер для отображения. Поэтому необходимо использовать преобразование выходов ИНС. Результаты оцениваются вероятностями и в дальнейшем не учитываются те из них, которые имеют вероятность меньше 0.5. Для сокращения избыточности оставшихся выходных значений использовался дополнительный критерий отбора.

Результат полученного решения – 2-3 кадра в секунду при обнаружении движущегося объекта. Данное решение является недостаточно производительным для безопасности производства. Чтобы повысить производительность обнаружения выполняется один раз в 50 кадров. В промежутке между обнаружениями используется алгоритм сопровождения объектов в видеопотоке на основе многомерной корреляционной фильтрации [5]. Использование алгоритма сопровождения объектов по видеопотоку позволило обновлять местоположение объекта для каждого кадра, затрачивая меньше вычислительной мощности. Использование значения координат центра, ширины, высоты позволило реализовать алгоритм идентификации объекта, основанный на координатах центра объекта. Это позволило распознать объект в видеопотоке.

Использование алгоритма сопровождения объектов в видеопотоке позволило обновлять местоположение объекта для каждого кадра, затрачивая меньше вычислительной мощности. Использование вышеописанного метода позволило достичь скорости в 12 кадров /секунда.

#### **Список использованных источников:**

1. Jetson Nano Developer Kit // NVIDIA AUTONOMOUS MACHINES [Электронный ресурс]. – Режим доступа: <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>. – Дата доступа: 20.03.2020.
2. Microsoft COCO: Common Objects in Context // Cornell University [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1405.0312/> Дата доступа: 20.03.2020
3. YOLOv3: An Incremental Improvement // Cornell University [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1804.02767/> Дата доступа: 20.03.2020
4. Accurate Scale Estimation for Robust Visual Tracking / Martin Danelljan [и др.]. BMVA Press, 2014.

# СПОСОБЫ РАСЧЁТА ТРАЕКТОРИИ ДВИЖЕНИЯ ТРАНСПОРТА

Арабей Д.И., Халиляев А.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Деменковец Д.В. – старший преподаватель

В докладе рассматриваются способы расчета траектории движения транспорта на основании данных получаемых с датчиков установленных на транспорте, а также алгоритмы быстро расширяющихся деревьев RRT и KD-tree.

В настоящее время системы активной безопасности транспортных средств широко применяются ведущими компаниями по производству автомобилей. Основным предназначением систем активной безопасности автомобиля является предотвращение аварийных ситуаций и помощь водителю в различных критических ситуациях. Также в данный момент во всём мире ведутся активные работы, направленные на создание автономных транспортных средств и различных робототехнических комплексов на их основе [1].

На сегодняшний день существуют различные алгоритмы и способы расчёта траектории движения. Для получения расчета траектории движения транспортных средств используются данные со следующих устройств: датчиков положения (угол поворота рулевого колеса), парковочных датчиков (установлены в переднем и заднем бамперах) и видеорегистратора. На основе данных с этих устройств можно рассчитать текущую траекторию движения и предполагаемую, при наличии препятствий. Для расчёта текущей траектории используются данные с датчиков положения. Для расчёта предполагаемой траектории с учётом имеющихся препятствий используется алгоритм быстро расширяющихся деревьев RRT [2].

Входными параметрами алгоритма являются начальное положение, карта местности, конечное положение. Информация о препятствиях составляется исходя из данных полученных с видеорегистратора. Для обработки данных с камеры необходимо разбить видеопоток на кадры для удобства обработки. Для этих целей используется библиотека OpenCV [3]. Данная библиотека подходит для систем реального времени.

Формула расчета угла поворота для текущего положения транспортного средства выглядит следующим образом:

$$\varphi_i = \varphi_{i-1} + \Delta\omega * K_{\pi} * \sigma, \quad (1),$$

где  $\varphi_i$  – величина угла поворота в текущем положении,  $\varphi_{i-1}$  – величина угла поворота в начальном положении,  $\Delta\omega$  – величина, показывающая необходимую корректировку значения угла, полученная из алгоритма RRT с учётом карты препятствий,  $K_{\pi}$  – корректирующий коэффициент, полученный исходя из данных парковочного радара,  $\sigma$  – значение, показывающее направление изменения угла [2].

RRT алгоритм предполагает построение карты препятствий. На выходе алгоритма формируется пройденный транспортным средством путь. Алгоритм KD-tree используется для эффективного нахождения следующей, максимально близкой от текущего положения точки в пространстве. Таким образом строится быстро расширяющееся дерево. После того как будет найдена максимально подходящая следующая точка, будет проведена проверка на достижимость. Если точка будет считаться достижимой, далее будет проводится проверка на то, возможен ли поворот и допустимо ли будет после поворота переконфигурировать карту движения.

Алгоритмы RRT и KD-tree используются для построения положений по всей траектории движения. Так как на карте могут присутствовать другие движущиеся объекты, алгоритм используется чаще, за счёт более частых обновлений карты [3].

Для повышения эффективности алгоритма используется также модификации для удаления промежуточных вершин на карте, ориентирование на точку финиша. Это позволяет проверить возможность прямого соединения последней точки, найденной алгоритмом RRT [2]. Удаление промежуточных вершин осуществляется для участков, на которых можно сделать траекторию прямой.

Таким образом, этот способ и алгоритмы могут эффективно применяться для расчета предполагаемой траектории движения транспорта для эффективного управления им.

## Список использованных источников:

1. Системы современного автомобиля. // Системы автомобиля [Электронный ресурс]. – Режим доступа: <http://systemsauto.ru/> Дата доступа: 20.03.2020
2. Модификация алгоритма RRT для определения оптимальной траектории движения автомобиля при объезде препятствий/ Ахметзянова И.З, Ионов М.А., Карабцев В.С. // Научная статья по специальности "Компьютерные и информационные науки", 2017. – С.148-154.
3. OpenCV (Open Source Computer Vision Library) // OpenCV [Электронный ресурс]. – Режим доступа: <http://opencv.org/> Дата доступа: 20.03.2020



## ПРОГРАММНОЕ СРЕДСТВО РЕАЛИЗАЦИИ ПРЕОБРАЗОВАНИЯ ГИЛЬБЕРТА-ХУАНГА

Бондарева Т.О., Пикиреня П.И., Леванцевич В.А

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Леванцевич В.А., м.т.н., ст. преподаватель

Рассмотрены классические способы анализа вибрационных сигналов. Указаны их основные недостатки. Исследуется возможность применения преобразования Гильберта-Хуанга для анализа вибросигналов. Разработан прототип программного средства, который выполняет декомпозицию исходного вибросигнала на эмпирические моды.

В настоящее время для обработки вибрационных сигналов в большинстве случаев применяются методы спектрального анализа, основанные на дискретном преобразовании Фурье, сущность которого состоит в том, что исходный сигнал во временной области, раскладывается на гармонические составляющие во временной области [2]. Однако существуют некоторые ограничения спектрального Фурье-анализа: исследуемая система должна быть линейной; анализируемые данные должны быть или периодическими или стационарными, т.е. амплитуда и частота отдельных составляющих сигнала должны быть постоянны во времени. Значительное количество реальных систем не удовлетворяют этим требованиям,

Недостатки спектрального преобразования могут быть частично устранены использованием вейвлет-преобразования, которое представляет собой свертку исследуемого сигнала с базисной вейвлет-функцией [2]. Основная проблема вейвлет преобразования-это выбор базисной функции.

Основой преобразования Гильберта-Хуанга является эмпирическая модовая декомпозиция исследуемого сигнала, в результате которой исходные данные разлагаются на ряд компонент, которые называются эмпирическими модами, а затем к полученному разложению применяется преобразование Гильберта [3].

Для реализации эмпирической модовой декомпозиции на первом этапе необходимо определить все максимальные и минимальные значения исследуемого вибросигнала  $y(t)$ .

На втором этапе по полученным минимальным и максимальным значениям строится верхняя и нижняя огибающие  $Y_{max_i}(t)$  и  $Y_{min_i}(t)$ . Для построения огибающей, как правило, используются кубические сплайны.

На третьем этапе вычисляется среднее значение огибающих по формуле:

$$m_1(t) = \frac{Y_{max_i}(t) + Y_{min_i}(t)}{2} \quad (1)$$

На четвертом этапе определяется первое приближение к первой функции эмпирической моды сигнала  $y(t)$ :

$$h_1(t) = y(t) - m_1(t) \quad (2)$$

На следующей итерации, принимая вместо функции  $y(t)$  функцию  $h_1(t)$  и повторяя этапы с первого по четвертый, находим второе приближение к первой функции моды  $h_2(t)$ . Аналогичным способом находим следующие приближения к первой функции моды. Итерации продолжают до тех пор, пока нормализованная квадратичная разность между двумя приближениями  $h_{i-1}(t)$  и  $h_i(t)$ , не станет меньше некоторого предельного значения:

$$\alpha = \sum_{i=0}^T \left( \frac{(h_{i-1}(t) - h_i(t))^2}{h_{i-1}(t)} \right) \quad (3)$$

Последнее значение  $h_i(t)$  представляет первую эмпирическую моду  $c_1(t) = h_1(t)$ . Далее первая эмпирическая мода вычитается из исходного сигнала:

$$y(t) - c_1(t) = r_1(t) \quad (4)$$

Для получения второй эмпирической моды над остатком  $r_1(t)$  так же выполняются этапы преобразования с первого по четвертый. Просуммировав полученные значения  $c_i(t)$  и последний полученный остаток  $r_n(t)$ , который может быть трендом или постоянной составляющей сигнала, получим разложение  $y(t)$  в виде

$$y(t) = \sum_{i=1}^n c_i(t) + r_n \quad (5)$$

Процесс определения новых эмпирических мод может быть остановлен в следующих случаях:

- сумма модовых функций, полученная по выражению (5) отличается от исходного сигнала на величину, которая не превышает заранее установленного значения;
- остаточная функция  $r_n(t)$  становится монотонной функцией без экстремумов.

Для анализа вибрационных сигналов с помощью преобразования Гильберта-Хуанга было разработано программное средства, которое позволяет проводить эмпирическую модовую декомпозицию вибрационных сигналов, отображать исходный сигнал и полученные моды (рисунок 1).

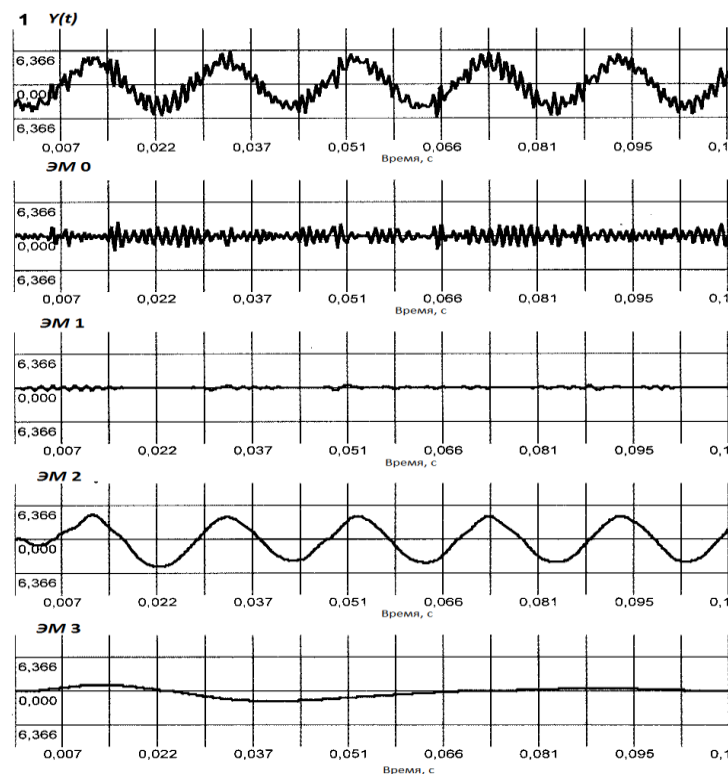


Рисунок 1 - Результат работы программного средства

На рисунке видно, что нулевая мода ЭМ 0 представляет собой наиболее высокочастотную часть сигнала, оставшиеся моды ЭМ 1 – ЭМ 3 отражают низкочастотную часть сигнала.

Выводы. Преобразование Гильберта–Хуанга позволяет определить зависимость амплитуды и частоты составляющих модовых функций сигнала от времени, что позволяет применить его для анализа нелинейных и нестационарных данных.

**Список использованных источников:**

1. Бранцевич, П.Ю. Применение разложения вибрационных сигналов на периодическую и шумоподобную составляющие. / П.Ю. Бранцевич, В.Э. Базаревский, С.Ф. Костюк // *Механика 2011: сб. науч. тр. V Белорусского конгресса по теорет. и прикладной механике*, Минск, 26–28 окт. 2011 г.:
2. Бурнаев Е.В. Применение вейвлет преобразования для анализа сигналов: Учебно-методическое пособие – М.: МФТИ, 2007, 138 с.
3. 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

# АНАЛИЗ СТРУКТУР НЕЙРОННЫХ СЕТЕЙ ENCODER-DECODER ДЛЯ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ НА МОБИЛЬНЫХ УСТРОЙСТВАХ

Бычик Юрий Григорьевич

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Ярмолик Вячеслав Николаевич – доктор технических наук

В данном докладе рассмотрены архитектуры нейронных сетей UNet и LinkNet класса Encoder-Decoder, позволяющие использовать сверточные сети для классификации изображений для решения задачи сегментации. Выполнена оценка применимости нейронных сетей вида Encoder-Decoder для запуска на мобильных устройствах. Описана нейронная сеть MobileNetV2 для классификации и способ ее использования в архитектурах UNet и LinkNet. Отражены экспериментальные результаты по оценке скорости работы нейронных сетей на мобильном устройстве.

Наиболее популярным классом нейросетевых методов для решения задач компьютерного зрения являются сверточные сети. Оператор свертки является основным вычислительным блоком. Последовательное применение свертки с точки зрения нейронных сетей позволяет выделять признаки “низкого уровня” (линии, точки, и т.п.) и далее аккумулировать признаки “низкого уровня” в признаки “высокого уровня” (контуры объектов и т.п.). В цифровом представлении найденные признаки представляют собой многомерные тензоры и используются для классификации изображений [1].

Аккумулированные признаки можно “трансформировать” в сегментированные области на исходном изображении с помощью некоторого обратного преобразования. Использование нейронных сетей для классификации в качестве основы для сегментации представляет собой архитектуру класса Encoder-Decoder. Наиболее часто используемыми архитектурами Encoder-Decoder являются UNet и LinkNet. Для работы UNet и LinkNet на мобильных устройствах необходимо использовать адаптированную для мобильных устройств сеть для классификации MobileNetV2 [2].

Скорость работы архитектур Encoder-Decoder линейно зависит от скорости работы сети для классификации MobileNetV2, которая используется в качестве Encoder-части. Коэффициент линейности зависит от скорости обратных преобразований в Decoder-части, как правило от количества сверток. Таким образом, нейронные сети типа Encoder-Decoder для сегментации на основе MobileNetV2 применимы для мобильных устройств [3].

MobileNetV2 состоит из начального слоя с 32 сверточными фильтрами и 17 остаточных блоков (residual blocks, bottlenecks). Каждый блок выполняет “распаковку-сжатие” признаков. На вход блока поступают трехмерные тензоры с  $N$  каналами. Далее идет оператор распаковки (expansion), где с помощью набора  $6N$  сверточных фильтров размерностью  $1 \times 1 \times N$  количество каналов увеличивается до  $6N$ , и на выходе каждой свертки дополнительно используется нелинейная функция Relu6. Далее применяются  $6N$  сверточных фильтров размерностью  $3 \times 3 \times 1$  с разделением по глубине (depthwise convolution). К результату каждого фильтра применяется нелинейная функция Relu6. На последнем этапе происходит “сжатие” количества каналов тензора с помощью  $M$  сверток размерности  $1 \times 1 \times 6N$  до размерности глубины  $M > N$ , где  $M \sim$  в 1,5 - 2 раза больше  $N$ . Для некоторых остаточных блоков  $M$  равно  $N$  для совпадения входных и выходных тензоров. Это позволяет совместить результаты работы остаточного блока с входными данными для лучшего обучения сети. На предпоследнем этапе получается тензор малой размерности с большим количеством каналов  $D$ , который сжимается последним оператором до размерности  $1 \times 1 \times D$ . Далее используется полносвязный слой с  $D$  входами и  $R$  выходами, которые представляют собой вероятности искомого класса. Описание операторов MobileNetV2 представлено в таблице 1.

Таблица 1 – Операторы MobileNetV2

Размер тензора, вход	Оператор	Кол-во каналов, выход	Кол-во операторов
224x224x3	Conv 2D 3x3	32	1
112x112x32	Bottleneck	16	1
112x112x16	Bottleneck	24	2
56x56x24	Bottleneck	32	3
28x28x32	Bottleneck	64	4
14x14x64	Bottleneck	96	3
14x14x96	Bottleneck	160	3
7x7x160	Bottleneck	320	1
7x7x320	Conv 2D 1x1	1280	1
7x7x1280	Average pooling, 7x7	1280	1
1x1x1280	Conv2d 1x1	R	1

Для обработки входного изображения размером 224x224x3 выполняется ~600 миллионов операций с плавающей точкой (FLOPS), нейронная сеть содержит ~3.5 миллионов параметров.

В таком виде у сегментации есть проблемы с определением границ. Это связано с тем, что признаки “высокого” уровня, полученные в результате работы сети, не учитывают локальный контекст каждого признака, т.е детали связи некоторой области изображения с соседними областями. Для решения этой задачи используются архитектуры UNet[4], LinkNet[5], которые на каждом этапе увеличения высоты и ширины тензора добавляют к данному тензору каналы из этапов Encoder-части. В UNet архитектуре используется операция concatenate, которая увеличивает общее количество каналов на каждом этапе работы Decoder-части. В архитектуре LinkNet используется операция add, которая складывает значения соответствующих каналов в связанных тензорах Encoder- и Decoder-частях. Схемы UNet и LinkNet приведены на рисунке 1.

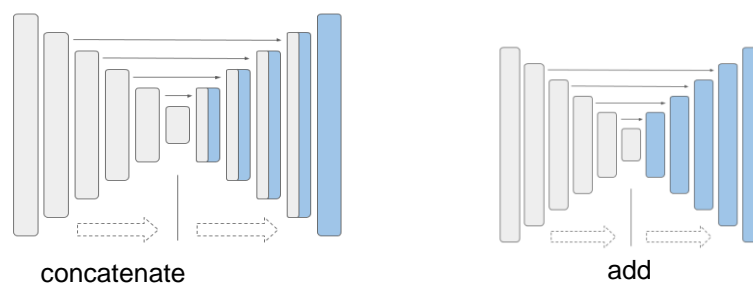


Рисунок 1 – Архитектуры Encoder-Decoder: UNet (слева), LinkNet(справа)

Для оценки скорости работы архитектур UNet и LinkNet с использованием MobileNetV2 разработано программное средство на языке Python с использованием фреймворка TensorFlow 2 [6]. С помощью данных инструментов выполнялось моделирование архитектур сетей UNet и LinkNet. Для тренировки исходной нейронной сети MobileNetV2 был использован набор данных ImageNet. Для тестирования скорости работы сетей на мобильных устройствах использовалась платформа Android, фреймворк TensorFlow Lite, 8-ядерный процессор Snapdragon 845 с частотой 2GHz. Экспериментальные результаты представлены в таблице 2.

Таблица 2 – Экспериментальная оценка скорости сегментации 1 изобр. с помощью UNet и LinkNet

Нейронная сеть	Скорость сегментации 1 изобр., мс.	Отношение к MobileNetV2
MobileNetV2	~70	1
UNet + MobileNetV2	~350	~5
LinkNet + MobileNetV2	~150	~2

Скорость работы сети для сегментации изображений больше, чем классификации. Однако данные архитектуры применимы для мобильных устройств, т.к. абсолютные значения скорости работы приемлемы для решения классов прикладных задач, не связанных с реальным временем.

**Список использованных источников:**

1. Гонсалес, Р. Цифровая обработка изображений, издание 3-е, исправленное и дополненное / Гонсалес Р., Вудс Р. – Изд. 3-е. – М. : Техносфера, 2012. – 1104 с.
2. Goodfellow I. Deep Learning / Goodfellow I., Bengio J., Courville A. – MIT Press, 2016 – 800 p.
3. Sandler M. MobileNetV2: Inverted Residuals and Linear Bottlenecks / Sandler M., Howard A.G., Zhu M. - 2018 IEEE / CVF Conference on Computer Vision and Pattern Recognition, 2018. – P. 4510-4520
4. Ronneberger O. U-Net: Convolutional Networks for Biomedical Image Segmentation / Ronneberger O, Fischer P., Brox T. - CoRR, vol. abs/1505.04597 – 2015
5. Chaurasia A. LinkNet: Exploiting encoder representations for efficient semantic segmentation / Chaurasia A, Culurciello E. - 2017 IEEE Visual Communications and Image Processing (VCIP), 2017 – P.1-4
6. Gulli A. Deep Learning with TensorFlow 2 and Keras: Regression, ConvNets, GANs, RNNs, NLP, and more with TensorFlow 2 and the Keras API / Guill A., Kapoor A. - Packt Publishing, 2019 – 646 p.

# ПРИМЕНЕНИЕ МЕТОДА ПОИСКА КЛЮЧЕВЫХ СЛОВ ДЛЯ ЗАДАЧИ ИНДЕКСИРОВАНИЯ ДОКУМЕНТОВ

*Вабищевич Д.П.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Парамонов А.И. – канд. техн. наук, доцент*

В работе предлагается подход к организации индексирования документов, для организации поиска по ним с ранжированием результатов. В основе положено применение алгоритма TextRank и дополнительная предобработка текстовых документов. В качестве индекса предлагается использовать вектор ключевых слов документной базы.

Значительное увеличение цифровых документов требует новых решений целого ряда задач, связанных с организацией документов и в том числе задачи поиска информации. Для её решения были разработаны алгоритмы от простейшего поиска по ключу (прямое нахождение слов запроса в документах) до построения моделей документов [1] и последующего сравнения их с запросом. В данной работе предлагается решение задачи индексирования документов, которое даст возможность поиска по документной базе с ранжированием результатов и сопоставления документов на схожесть.

В качестве результата операции индексирования должно получаться некоторое представление документа, которое будет отражать его основное содержание. В качестве такого представления документа предлагается использовать вектор по универсуму терминов документной базы. Терминами выступают извлеченные из документов ключевые слова и фразы. Размерность вектора соответственно равна общему числу терминов, которые получены из документов на данный момент.

Для процесса ранжирования необходимо определить алгоритм вычисления меры схожести документов (а точнее их представлений) друг с другом и с поисковым запросом. Для идентичности операции сравнения с представлениями документов, запрос также преобразуются в аналогичный вектор терминов. Он будет содержать информацию о присутствии ключевых слов в запросе. Вектора проиндексированных текстов помещаются в  $k$ -мерное дерево, так как данная структура данных обеспечивает быстрый поиск и ранжирование по набору  $k$ -мерных векторов. Для задачи ранжирования в качестве меры схожести запроса и документа предлагается использовать метрику Минковского с  $p=1$  (расстояние городских кварталов,  $L_1$  норма). В нашем случае оно будет равно количеству терминов, относящихся только к запросу и только к документу. Таким образом, тексты с меньшим значением расстояния более похожи на запрос по смыслу. Следует учесть, что при добавлении новых текстов  $k$ -мерное дерево разбалансируется, что уменьшает скорость поиска и ранжирования. Также с каждым добавлением нового документа может изменяться и размерность вектора индексирования. Все это приводит к необходимости перестроения дерева в течение работы модуля, при накоплении определенного количества новых документов. Вопрос частоты балансировки дерева требует дополнительного исследования и будет рассмотрен в дальнейшем.

Извлечение ключевых слов предлагается выполнять с помощью метода поиска ключевых слов TextRank [2]. Принцип его работы заключается в построении графа со словами в вершинах и подсчете на его основе для каждого слова числового рейтинга. Он показывает, насколько слово часто встречается в этом тексте в разных контекстах. Чем больше это число, тем вероятнее, что слово важно для текста и отражает суть его содержания. Основной плюс алгоритма в том, что для извлечения ключевых слов он использует только статистики слов в текущем документе. Это значит, что он не требует хранения никакой дополнительной информации.

Для улучшения качества результатов работы алгоритма к тексту применяется предобработка. Во-первых, из текста убирается пунктуация и служебные слова, которые сами по себе не несут смысловой нагрузки (предлоги, местоимения и т.п.). Затем применяется алгоритм нахождения устойчивых словосочетаний, который основывается на подсчете того, как чаще слова встречаются вместе или по отдельности [2]. В результате работы этого алгоритма часть ключевых слов (словосочетания) объединяются в один токен, и TextRank работает с ними уже как с одним термином.

Предложенный подход обеспечивает хорошее качество индексирования документной базы и как следствие быстрого поиска с ранжированием по ней. При этом для начала работы не требуется предварительно научения и наборов доменно-специфичных текстов, что делает качество работы независимым от доменной области и языка, а также от количества хранимых в базе текстов.

#### **Список использованных источников:**

1. Парамонов А.И. Представление знаний гибридной моделью для систем интеллектуального поиска / Вестник Донецкого национального университета – 2005. – Серия А, №1. – С. 404-409.
2. R. Mihalcea and P. Tarau. TextRank: Bringing Order into Texts// Proc. of the 9th Conf. on Empirical Methods in Natural Language Processing. – 2004. – С. 404-411.
3. Distributed Representations of Words and Phrases and their Compositionality / T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean // In Advances in Neural Information Processing Systems 26, pages 3111-3119, 2013. – 9 с.

## ИССЛЕДОВАНИЕ СРЕДСТВ СОЗДАНИЯ И РЕАЛИЗАЦИИ ЭЛЕКТРОННЫХ ЦЕННЫХ БУМАГ КОМПАНИИ НА ОСНОВЕ ТЕХНОЛОГИИ БЛОКЧЕЙН

*Вариков К.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Фадеева Е.Е. – ассистент*

Вопросом как сохранить доход с течением времени человечество задавалось с тех самых пор, как появилась торговля. На сегодняшний день наилучшим способом является инвестиции в различные компании. Целью проекта является улучшение данного процесса путём снижения издержек с помощью применения современных технологий.

На протяжении развития человечество тестировало различные способы сохранения своего труда. И неважно, был это какой-либо продукт или деньги – всё это лишь эквивалент нашего труда. На сегодняшний день существует три основных способа сохранения: откладывание «под подушкой», банковские вклады, инвестиции. Первые два способа сильно подвержены таким явлениям, как: инфляция, девальвация, деноминация, дефолт.

Изучение процесса инвестиций в различные компании и/или покупка долговых обязательств различных стран показал следующее: крупные международные компании показывают средний рост 7-12% годовых, а компании в РБ показывают куда более внушительный рост. Глядя на показатели становится очевидно, что вариант инвестиций является предпочтительным среди прочих вариантов сохранения/приумножения денежных средств. Однако, в этом процессе существует много издержек. Для инвестора – это высокий порог входа и комиссия посредников в лице банка или брокера. Для предпринимателя количество издержек ещё больше. Для выхода на биржу необходимо воспользоваться услугами андеррайтера [2]. Что зачастую является попросту невозможным для мелкого и средних бизнесов. Различные биржи также имеют различные пороги входа (крупнейшая Нью-Йоркская биржа – капитализация компании от 50 миллионов долларов).

Таким образом, были сформированы следующие требования к средству создания и реализации электронных ценных бумаг компаний:

- 1) Предоставить возможность мелким инвесторам самостоятельно покупать ценные бумаги различных компаний;
- 2) Предоставить возможность предпринимателям выпускать свои электронные ценные бумаги, самостоятельно устанавливая правила выплат дивидендов своим инвесторам.
- 3) Обеспечить максимальную безопасность как для инвестора, так и для предпринимателя.
- 4) Обеспечить невозможность обмана инвесторов (эмиссия новых ценных бумаг, невыполнение условий сделки и т.п.)

К счастью, технологии так же не стоят на месте и сегодня все вышеперечисленные требования выполнимы с технической точки зрения. Технология блокчейн позволяет организовать все операции с минимальным количеством посредников. При этом все операции являются прозрачными, что позволяет в той или иной степени минимизировать риски кражи, потери, обмана и т.п. Платформа Ethereum приносит в мир блокчейн Смарт-контракты, которые являются аналогами своих бумажных прародителей. Это значит, что всё то, что можно задекларировать на бумаге можно реализовать в электронном виде. Однако, они лишены возможности обмана, т.к. весь исходный код смарт-контрактов хранится в блокчейне, а он является неизменяемой(immutable) сущностью и полностью прозрачным, что позволяет любому человеку посмотреть условия сделки с той или иной компанией. Так же, любой человек может бесплатно создать кошелек в блокчейне Ethereum и приобрести электронные ценные бумаги той или иной компании, находящейся в данном блокчейне.

Алгоритм взаимодействия с проектом будет примерно следующим.

- 1) Предприниматель связывается с нами и рассказывает о том, как бы он хотел привлечь инвестиции в свой бизнес.
- 2) Мы реализуем смарт-контракт, максимально подходящий для его целей.
- 3) Далее этот смарт контракт попадает в блокчейн.
- 4) Потенциальный инвестор заходит на веб-сайт, где он может приобрести электронные ценные бумаги компаний, продать их, посмотреть котировки, посмотреть подробную информацию о компании.

Первое что нужно сделать – это разработать смарт-контракт. Для этих целей используется специальный объектно-ориентированный, предметно-ориентированный язык программирования самовыполняющихся контрактов – Solidity. Данный язык является строго типизированным, что позволяет минимизировать количество ошибок, которые могут быть допущены разработчиком. Далее полученный смарт-контракт необходимо протестировать. Тестирование является крайне важным

этапом, т.к. блокчейн является неизменяемой сущностью. Первый этап тестирования проводится с помощью симулятора блокчейн – Ganache. Данный инструмент позволяет развернуть локальную сеть блокчейн на локальном компьютере. Второй этап тестирования – это загрузка отдельного смарт-контракта в тестовую сеть блокчейн, например, Rinkeby. И только после того, как смарт-контракт был протестирован в данной сети, можно загружать его в так называемую «MainNetwork».

Далее необходимо разработать пользовательский интерфейс для взаимодействия со смарт-контрактом. Для этого будет разработано SPA (Single Page Application) веб-приложение, чтобы предоставить максимально удобный опыт взаимодействия с проектом.

Серверная часть приложения будет построена на базе платформы NodeJS с использованием объектно-реляционной системы управления базами данных - PostgreSQL. Задача сервера будет предоставлять информацию о компаниях, заинтересованных в инвестициях, графики изменения цены токенов и т.п. Все финансовые операции происходят независимо от серверной части. В браузере пользователя с помощью расширения Metamask. Данное расширение позволяет подключить личный кошелек пользователя к браузеру, что в свою очередь позволяет веб-приложениям определять браузер пользователя, как браузер подключенный к блокчейн сети. Если у пользователя обнаружен подключенный к сети кошелек, то он имеет возможность приобрести токены, понравившейся компании.

Таковы основные идеи проекта для создания и реализации электронных ценных бумаг компании.

**Список использованных источников:**

1. MyFin.by [Электронный ресурс] URL: <https://myfin.by/wiki/term/infljaciya>.
2. Banki.ru [Электронный ресурс] URL: [https://www.banki.ru/wikibank/anderryting\\_na\\_ryinke\\_tsennyih\\_bumag/](https://www.banki.ru/wikibank/anderryting_na_ryinke_tsennyih_bumag/)

## ПРОГРАММНОЕ СРЕДСТВО ОЦЕНКИ ДЕМПИРУЮЩИХ СВОЙСТВ ПОКРЫТИЙ

*Верещагин Н. В., Буняк Г. С., Леванцевич В.А*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

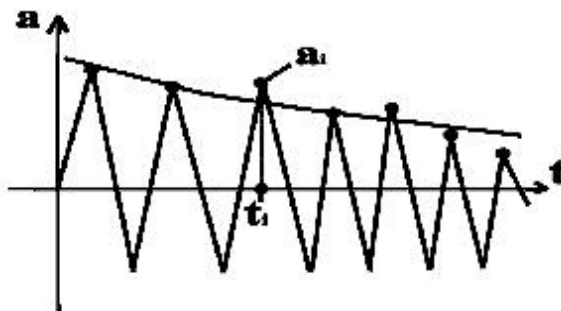
*Деменковец Д.В., м.т.н., ст.преподаватель*

Рассматривается программная реализация метода оценки демпфирующих свойств покрытий по логарифмическому декременту затухания. Описан алгоритм аппроксимации огибающей виброграммы свободных колебаний экспоненциальной функцией с помощью метода наименьших квадратов.

В настоящее время, для борьбы с вибрациями и шумами в механических системах, ведутся интенсивные исследования по разработке специальных, вибропоглощающих покрытий, нанесение которых на узлы и детали машин позволяет значительно уменьшить вибрации и шумы [1].

На практике, для оценки вибропоглощающих свойств покрытий наибольшее распространение получил метод, основанный на измерении параметров затухающих колебаний. При этом за основную метрику оценки свойств покрытий по величине поглощенной энергии механических колебаний, принимают логарифмический декремент затухания колебаний[2].

Точность определения логарифмического декремента по виброграмме зависит от точности измерения амплитуд колебаний. Как показали экспериментальные исследования на практике, в силу случайных факторов, наблюдается разброс значений измеряемых амплитуд [3]. Для уменьшения погрешности определения декремента необходимо аппроксимировать огибающую виброграммы.(рис.1).



1).

Рисунок 1 – Аппроксимация огибающей виброграммы

Чаще всего для аппроксимации свободных, затухающих колебаний используются экспоненциальные функции вида[4]:

$$y = Ae^{-kt} . \quad (1)$$

Пусть даны экспериментально полученные данные затухающего колебательного процесса в виде множества пар значений  $(a_i, t_i)$ , где  $a_i$  - максимальные значения амплитуд циклов колебаний,  $t_i$  соответствующие им значения времени,  $i = 1, n$ , а также аппроксимирующая функция вида (1). Для построения огибающей методом наименьших квадратов прологарифмируем левую и правую части выражения (1):

$$\ln y = \ln A + (-k)t . \quad (2)$$

Для удобства расчетов перепишем выражение в виде:

$$Z = C - kt , \quad (3)$$

где  $Z = \ln y$ ,  $C = \ln A$ .

Тогда по методу наименьших квадратов:

$$f(C, k) = \sum_{i=1}^n (Z_i - (C - kt_i))^2 \rightarrow \min \quad (4)$$

Для нахождения неизвестных  $C, k$  продифференцируем (4) по  $C, k$ :

$$\begin{cases} \frac{\partial f}{\partial C} = 2 \sum_{i=1}^n (Z_i - C + kt_i)(-1) \\ \frac{\partial f}{\partial k} = 2 \sum_{i=1}^n (Z_i - C + kt_i)t_i \end{cases} , \quad (5)$$

Приравняв к нулю частные производные получим систему линейных уравнений :

$$\begin{cases} \sum Z_i \cdot t_i - C \cdot \sum t_i + k \sum t_i^2 = 0 \\ - \sum Z_i + C \cdot n - k \sum t_i = 0 \end{cases} \quad (6)$$

Решим линейную систему уравнений относительно  $C$  и  $k$ :

$$C = \frac{-\sum Z_i \cdot \sum t_i^2 + \sum t_i \cdot \sum t_i \cdot Z_i}{-n \cdot \sum t_i^2 + (\sum t_i)^2} , \quad (7)$$

$$k = \frac{n \sum Z_i t_i - \sum Z_i \cdot \sum t_i}{-n \sum t_i^2 + (\sum t_i)^2} , \quad (8)$$

Зная коэффициент  $C$  по формуле (8) можно определить значение коэффициента  $A$ :

$$A = e^C , \quad (9)$$

Используя значения  $A, C, k$ , по формуле (1) строится аппроксимирующая огибающая виброграммы.

Алгоритм определения огибающей состоит из следующих основных шагов:

- 1) открывается файл с текущими парами значений  $(a_i, t_i)$ -амплитуда, время,  $i = 1..n$ );
- 2) вычисляется сумма значений времён  $t_i$ ;



- 3) вычисляется сумма квадратов времён  $t_i^2$ ;
- 4) определяется сумма значений логарифмов амплитуд  $\ln a_i$ ;

- 5) вычисляется сумма значений произведений  $\sum_{i=1}^n t_i \cdot \ln a_i$ ;

6) по формулам (7, 8, 9) определяется значение коэффициентов  $A, C, k$ ;

7) по формуле (1) вычисляются значения огибающей виброграммы;

8) по огибающей виброграммы определяется декремент затухания.

По предложенному алгоритму, было разработано программное средство построения огибающей виброграммы, и вычисления логарифмического декремента затухающих колебаний.

Выводы. Использование алгоритма и программного средства позволяет:

- повысить точность оценки демпфирующих свойств покрытий по величине поглощения энергии колебаний;
- сократить затраты времени на исследование свойств покрытий.
- программное средство может быть использовано для автоматизации научного эксперимента при исследовании свойств покрытий.

**Список использованных источников:**

1. Белоцерковский М.А., Рудько В.М. Влияние напыленных полимерных покрытий на затухание колебаний. Вестник Полоцкого гос. университета, № 12, 2017 г. с. 71.
2. Рахштадт А.Г. Современные методы испытаний материалов в машиностроении. – М.: Наука, 2001, 24 с.
3. Чернышев В.М. Методы исследования вибропоглощающих свойств материалов в условиях сложного нагружения / Волгоград.гос.техн. университет. – Волгоград, 2000. Деп. в ВИНТИ 15.03 2000, № 624-В94
4. Кассандров О.Н., Лебедев В.В. Обработка результатов наблюдений. –М.: Наука, 2016.

## МЕТОДЫ ОБФУСКАЦИИ ПРОГРАММНЫХ СРЕДСТВ С ЦЕЛЬЮ ЗАЩИТЫ ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Водейко А.Э.*

*Ярмолик В.Н. – д.т.н., профессор*

На сегодняшний день широко используются технологии разработки ПО, сильно уязвимые к обратному проектированию. В первую очередь, эта проблема касается таких языков программирования, как Java, C#. Для затруднения обратного проектирования используется метод, называемый обфускацией, который не изменяет функциональность программы, но усложняет анализ и понимание алгоритмов работы программы и усложняет модификацию.

Обфускация состоит в отображении исходной программы  $P=\{..., S_j, ...\}$ , состоящей из исходного объекта  $S_j$ , в новую программу  $P^*=\{..., T_i(S_j), ...\}$  на основании эквивалентных преобразований  $T_i$  [1]. Сама по себе обфускация не позволяет пресечь попытки несанкционированного доступа к программе, так как не изменяет функциональность программы, поэтому её применяют с другими методами защиты. Целью обфускации является затруднение, а в идеале - предотвращение таких атак, как обратное проектирование и модификация программы. Различают следующие уровни обфускации, характеризующиеся целью, объёмом и сложностью преобразований кода: лексическая обфускация, обфускация потока выполнения, обфускация данных[2].

Наиболее простой является лексическая обфускация. Она заключается в снижении читабельности исходных кодов программного средства. Это достигается за счёт следующих действий:

- добавления лишних, изменение или удаление комментариев;
- нестандартного форматирования;
- переименование идентификаторов так, чтобы их названия не отражали их предназначения и трудно воспринимались человеком.

Такой метод имеет низкую эффективность, так как не изменяется схема работы программы и практически полностью восстанавливаются при автоматической деобфускации.

Обфускация данных использует реструктуризацию данных в изменении представления данных. Она осуществляется путём объединения нескольких массивов в один, или, наоборот, разделение одного массива на несколько отдельных переменных, использования одной переменной большей разрядности для хранения нескольких переменных меньшей разрядности, изменение количества измерений массива, расщепления переменных и т.д.

Обфускация потока выполнения заключается в преобразовании графа передачи управления программы.

Большинство алгоритмов преобразования управления используют понятие “непрозрачный

предикат”. Непрозрачным предикатом называют выражение, результат которого известен обфускатору, но является трудно определимым для деобфускатора. Непрозрачные предикаты делятся на три группы[3]:

- предикат, значение которого всегда “истина”. Обозначается PT.
- предикат, значение которого всегда “ложь”. Обозначается PF.
- предикат, значение которого зависит от некоторых условий. Обозначается P?.

С помощью трёх разновидностей предикатов можно выполнить преобразования, модифицирующие передачу управления в программе.

Для определения эффективности обфускации существуют две группы методов: аналитические и эмпирические[4]. Аналитические методы основываются на критериях устойчивости, эластичности и стоимости преобразования.

Устойчивость характеризует степень сложности обратного проектирования обфусцированного кода. Эластичность – степень защиты обфусцированного кода от использования программ-деобфускаторов. Стоимость преобразования показывает, насколько увеличилось потребление системных ресурсов после осуществления процесса обфускации. Аналитические методы подходят для сравнения разных алгоритмов обфускации, но не отвечают на вопрос, какой из алгоритмов подходит лучше всего к определённому коду.

Эмпирические методы основываются на статистических данных исследований применения разных алгоритмов и позволяет выяснить, какой алгоритм будет более эффективным для определённого кода. Таким образом, обфускация является важным методом защиты программного обеспечения от обратного проектирования, однако данные методы не получили широкого распространения на данный момент.

**Список использованных источников:**

1. 1. Криптография, стеганография и охрана авторского права / В.Н.Ярмолик, С.С. Портянко, С.В.Ярмолик / Издательский центр БГУ – Минск, 2007. – С. 190-195
2. On the (Im)possibility of Obfuscating Programs / B. Barak, O. Goldreich, R. Impagliazzo, S. Rudich, A. Sahai, S. Vadhan, K. Yang. - LNCS, 2010. - P. 1-18.
3. An approach to the obfuscation of control-flow of sequential computer program / S. Chow, Y. Gu, H. Johnson, V. Zakharov - LNCS, 2001. - P. 144-155.
4. Candidate indistinguishability obfuscation and functional encryption for all circuits / Garg S., Gentry C., Halevi S., Raykova M., Sahai A., and Waters B. FOCS, 2013 - P. 22-23 5. On best-possible obfuscation / Goldwasser S., and Guy N. R. - TCC, 2007.

## **ПРОГРАММНОЕ СРЕДСТВО МОДЕЛИРОВАНИЯ И ПОИСКА НЕИСПРАВНОСТЕЙ ЗАПОМИНАЮЩИХ УСТРОЙСТВ**

*Деменковец Д.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Ярмолик В.Н. – профессор, доктор технических наук*

В тексте доклада описывается программное средство моделирования и поиска неисправностей запоминающих устройств. Дается классификация неисправностей запоминающих устройств. Описываются функциональные возможности этого ПС при использовании маршевых тестов. Описываются получаемые программным средством оценки поиска неисправностей при маршевом тестировании.

Быстрое развитие полупроводниковых технологий привело к значительному увеличению объема и плотности оперативной памяти (ОЗУ, RAM) на одном чипе. Для тестирования памяти требуется больше времени. В связи с этим наблюдается повышенный интерес к проблемам диагностирования оперативной памяти (ОЗУ, RAM). Этот интерес связан с тем, что ОЗУ является главным компонентом современных вычислительных систем, таких как встроенных систем (system-on-a-chip, SoC) и сетей на кристалле (nets-on-a-chip). Для тестирования ОЗУ используются, как правило, тесты линейной длины семейства «Марш» («March») [1,2]. Актуальным является проверка эффективности маршевых тестов. Проверки полноты покрытия данных тестов. Оценка времени работы маршевых тестов. Проверка новых методов тестирования запоминающих устройств. Основной сложностью проверки является симуляция работы реальной памяти и поведение ячеек памяти при различных неисправностях. В докладе рассматриваются основные типы неисправностей запоминающих элементов, предложенных Ван де Гором оперативного запоминающего устройства (ОЗУ, RAM) [3].

Входными данными для программного средства моделирования являются маршевая последовательность или последовательности и последовательность адресов ячеек памяти (адресная последовательность). Выходными данными является: время, затраченное на тестирование, полнота

тестового покрытия, затраченное количество оперативной памяти компьютера, средняя загрузка процессора, оценка производительности, косвенная оценка энергоэффективности тестов.

Далее рассмотрим основные типы неисправностей запоминающих устройств, поддерживаемых проектируемым ПО. К неисправностям матрицы запоминающих устройств в первую очередь относят неисправности, в которых участвуют: одна ячейка; две ячейки; несколько ячеек, в общем случае более чем две, без ограничений на их количество. Общепринятыми являются следующие модели неисправностей ячеек памяти.

К неисправностям, затрагивающим одну ячейку ЗУ, относят [3]:

- Константные неисправности (stuck-at faults - SAF). Неисправный запоминающий элемент постоянно находится в состоянии логического нуля (s-a-0) или логической единицы (s-a-1), независимо от операций, выполняемых с неисправным элементом и другими элементами ЗУ. (уточнить, это с обратной связью, т.е. мы записываем, что-то в ответ получаем? Спросить многократные и однократные константные неисправности).

- Переходные неисправности (transition faults - TF). Подобные неисправности характеризуются невозможностью перехода состояния неисправного запоминающего элемента из 0 в 1 (TF $\uparrow$ ), или из 1 в 0 (TF $\downarrow$ ) при выполнении соответствующих операций записи. Если ячейка, имеет переходную неисправность, оказывается в состоянии, из которого она не может перейти в другое, то ее поведение повторяет поведение ячейки, содержащей константную неисправность. Данную неисправность не нужно путать с типом неисправности взаимного влияния, которая рассматривается ниже.

Среди неисправностей, в которых участвуют две ячейки ЗУ, выделяют следующие неисправности [3]:

- Неисправности взаимного влияния (coupling fault - CF). При описании данной неисправности выделяют влияющую ячейку (aggressor cell), определяемую ее адресом  $i$ , изменение логического состояния которой воздействует на состояние зависимой ячейки (victim cell) с адресом  $j$ .

- Инверсные неисправности взаимного влияния (inverse coupling faults - CF $\text{in}$ ). При наличии данной неисправности изменение значения  $b_i$  влияющей ячейки вызывает инвертирование значения  $b_j$  зависимой ячейки. Возможны четыре вида данного типа неисправностей CF $\text{in}$ .

- Неисправности взаимного влияния прямого действия (idempotent coupling faults - CF $\text{id}$ ). При изменении значения  $b_i$  влияющего запоминающего элемента происходит принудительная установка определенного логического значения 0 или 1 в зависимом запоминающем элементе. Различают восемь неисправностей прямого действия.

- Статические неисправности взаимного влияния (state coupling faults - CF $\text{st}$ ). Переход зависимой ячейки в какое-либо состояние  $b_j$  возможен при определенном значении  $b_i$  влияющей ячейки. Возможно восемь неисправностей CF $\text{st}$ .

Модели кодочувствительных неисправностей в данной статье не рассматриваются и данным программным средством не поддерживаются.

Среди множества тестов запоминающих устройств выделяют так называемые традиционные тесты, которые позволяют обеспечивать некоторую степень покрытия возможных неисправностей при соответствующих временных затратах. Для данного программного средства реализованы следующие широко известные тесты типа шахматная доска (checkerboard), тест бегущая 1/0 (walking 1/0), тест бегущая диагональ (sliding diagonal), тест пинг-понг (ping-pong), галопирующая 1/0 (galloping 1/0).

Во всех тестах запоминающих устройств операция чтения содержимого ячейки сопряжена с проверкой прочитанного значения 0/1 с заранее записанным значением 0/1 в данную ячейку.

Разрабатываемое программное средство поддерживает выполнение маршевых тестов (march tests). Маршевые тесты имеют линейную зависимость сложности  $O(N)$  от емкости  $N$  памяти. Данные тесты обладают рядом достоинств, среди которых приемлемая покрывающая способность неисправностей запоминающих устройств, описанных выше, а также простота реализации тестовой процедуры.

Данное программное средство моделирует только неисправности массива запоминающих элементов памяти, которые описаны выше, и на данном этапе не поддерживаются неисправности схем обрэмления, которые также относятся к неисправностям запоминающих устройств.

В данном программном средстве для удобства модель памяти представляется и рассматривается как двумерное пространство в виде матрицы с одинаковым количеством строк и столбцов. Входными тестовыми данными являются адреса запоминающих ячеек, состоящие из адресов по горизонтальной и вертикальной оси матрицы ячеек памяти

Моделирование константного типа (stuck-at faults - SAF) неисправности программным способом реализовано путем установки значения соответствующего свойства экземпляра объекта «ячейки памяти» с проверкой типа неисправности (соответствующего поля экземпляра объекта).

Симуляция поиска переходных неисправностей (transition faults - TF) реализованы схожим способом, с выполнением дополнительной проверки значения состояния ячейки (соответствующего свойства экземпляра объекта «ячейка памяти»). В зависимости от типа неисправности запись соответствующего состояния выполняется или остается неизменным.

Неисправности взаимного влияния реализованы с использованием паттерна (шаблона проектирования) «наблюдатель» с помощью ссылки на метод (делегата) [4].

Программная реализация позволяет в последующем легко расширять функциональные возможности и последующем позволит легко добавлять поддержку новых типов неисправностей, в том числе сложных и трудно обнаруживаемых. Данное программное средство позволяет симулировать значительные объемы оперативной памяти. Также ПС позволяет ручным способом организовать и проанализировать основные категории тестовые воздействия [3,5]: точечные тестовые наборы (point patterns), узкополосные тестовые наборы (strip patterns), блочные тестовые наборы (block patterns). В последующем на вход программы представить адресную последовательность (address set).

Программное средство специально разрабатывается на кафедре ПОИТ БГУИР для моделирования и верификации обнаруживающей способности проверяющих тестов на предмет диагностирования расширенного множества одиночных и кратных неисправностей.

Одним из основных критериев оценки эффективности тестов запоминающих устройств является время выполнения теста, которое зависит от количества операций обращения к тестируемому устройству и от времени отклика устройства. Обобщающая характеристика эффективности тестов называется сложность теста  $Q$ (тест). Эта характеристика оценивается как количество всех элементарных операций (чтения\запись), необходимых для реализации теста [6].

Под оценкой работы моделирования и поиска неисправностей запоминающих устройств понимается критерий качества маршевого теста, который определяется двумя параметрами:

- процент обнаруженных неисправностей в рамках принятой модели (процент покрытия);
- временные затраты на тестирование памяти, которое масштабируется количеством обращений к ячейкам памяти в процессе генерации теста в зависимости от ее емкости  $N$  (параметр  $N$  – количество ячеек памяти).

**Список использованных источников:**

1. Ярмолик, В. Н. Контроль и диагностика вычислительных систем: [монография] / В. Н. Ярмолик. – Минск : Бестпринт, 2019. – 387 с. : ил. 75.
2. Ярмолик, В. Н. Обзор методов неразрушающего тестирования ОЗУ / В. Н. Ярмолик, А. П. Занкович // Доклады БГУИР. - 2005. - № 4 (12). - С. 62 - 72.
3. Goog, A.J. Testing Semiconductor Memories, Theory and Practice / A.J. Googr. – UK, Chichester: John Wiley & Sons, 1991/ 536 p.
4. Тепляков С. Паттерны проектирования на платформе .NET : книга // С. Тепляков – СПб.: Изд-во: Питер, 2015. – 320 с.: ил.
5. Chan, F.T. Proportional sampling strategy guidelines for software testing practitioner / F.T. Chan, T.Y. Chen, I.K. Mak, T.T. Yu // Information Software Technology – 1996. – Vol.38. - №12. – P. 775-782.
6. Ярмолик, В. Н. Методы и средства встроенного самотестирования с пониженным энергопотреблением / В. Н. Ярмолик, И. А. Мурашко // Доклады БГУИР. - 2004. - № 1 (5). - С. 102 - 114.

## **ПРОГРАММНОЕ СРЕДСТВО АВТОМАТИЗАЦИИ ВЗВЕШИВАНИЯ ЖЕЛЕЗНОДОРОЖНЫХ ВАГОНОВ В ДИНАМИЧЕСКОМ РЕЖИМЕ**

*Деменковец Д.В., Дубицкая К.Ю.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Деменковец Д.В. – старший преподаватель*

В докладе рассматривается программное средство автоматизации взвешивания железнодорожных вагонов в режиме динамического режима (без остановки локомотива), а также алгоритм процесса фиксации веса вагонов на основе записанного видеоряда.

Транспортировка грузов по железнодорожным путям выгодно отличается безопасностью, надежностью, относительно низкой себестоимостью относительно других способов доставки. Большой популярностью становится взвешивание железнодорожных вагонов в режиме динамического взвешивания, т.е. без остановок локомотива для поочередного взвешивания каждого из вагонов [1]. Этот режим значительно ускоряет взвешивание железнодорожного состава в целом в сравнении со статическим взвешиванием. Также сокращается время работы составителей поездов и локомотива. Особенно на транзитных станциях, в том числе приграничных, где необходим только учет веса и важно время прохождения железнодорожных составов без простоев производства. Таким образом, автоматизация процесса взвешивания является актуальной задачей для многих предприятий, регулярно перевозящих сырье и продукцию посредством железнодорожного транспорта [2].

По способу взвешивания железнодорожные весы подразделяют на статические, динамические и

статодинамические. Статодинамические весы позволяют измерять вес вагонов в двух режимах, статическим и динамическом. При динамическом взвешивании необходимо обеспечивать некоторые условия, к которым относятся, постоянная скорость движения железнодорожного состава, отсутствие дефектов на поверхности колес вагонов, не допускается уклон рельсов на измерительном участке. Эти условия значительно сказываются на точности получаемых значений веса. Тем не менее при большом грузопотоке и невысокой стоимости взвешиваемых материалов целесообразно использование режима динамического взвешивания вагонов в движении [1].

Процесс взвешивания железнодорожных вагонов осуществляется программным средством входящим в состав программно-аппаратного комплекса [3]. Переход в режим динамического взвешивания осуществляется путем отправки на весоизмерительный индикатор ПК соответствующей команды. Алгоритм работы ПК в режиме динамического взвешивания состоит из следующих этапов:

- программный модуль ПК запрашивает у весоизмерительного индикатора параметры взвешивания с помощью протоколов TCP/IP интерфейса Ethernet;
- при проходе состава весопроецессор подсчитывает количество осей, прошедших через каждую платформу;
- по срабатыванию таймера, ПК запрашивает значения количества осей и при их изменении на ненулевое значение фиксируется прохождение железнодорожного состава и начинается запись видеоряда с внешних IP-камер, направленных на платформы;
- окончание записи видеоряда которое происходит после обнуления значения количества осей поступающего от весоизмерительного процессора;
- отправка команды на получение списка протоколов и получение текстового файла с данными полученными в результате динамического взвешивания;
- из полученного файла прочитываются точные времена нахождения вагонов на весовой платформе;
- формирование и сохранение в форматах Microsoft Word и Excel отчетов о взвешиваниях на основании видеоряда и меток времени, а также расчет специфических параметров конкретных грузов (например, объем сниженных газов) полученных от протоколов динамического взвешивания.

**Список использованных источников:**

1. Частное производственно-торговое унитарное предприятие «Саха-пром» - Автоматизация взвешивания железнодорожных вагонов [Электронный ресурс] ~/ --- Режим доступа: <http://sakha-prom.by/smart/avtomatizatsiya-vzveshivaniya.html> --- Дата доступа: 17.03.2020.
2. Деменковец Д.В., Дубицкая К.Ю. Алгоритм автоматического взвешивания железнодорожных вагонов / Д. В. Деменковец, Дубицкая К.Ю. // Компьютерные системы и сети: 55-я юбилейная конференция аспирантов, магистрантов и студентов учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», 22-26 апреля 2019 г., БГУИР, Минск, Беларусь: тезисы докладов. – Мн. – 2019. – 287 с.; ил. С. 94-95. 3. Цитраты алюминия (III) / В.В. Чевела [и др.] // Ученые записки казанского университета: Естественные науки, 2011. – С.61-69 .
3. Деменковец, Д. В. Программное средство автоматизации процесса динамического взвешивания железнодорожных вагонов с распознаванием номеров на основе искусственной нейронной сети / Деменковец Д. В., Дубицкая К. Ю., Борисевич А. Н. // Информационные технологии и системы 2019 (ИТС 2019) = Information Technologies and Systems 2019 (ITS 2019) : материалы международной научной конференции, Минск, 30 октября 2019 г. / Белорусский государственный университет информатики и радиоэлектроники; редкол. : Л. Ю. Шилин [и др.]. – Минск, 2019. – С. 38 – 39.

## **ПРАКТИЧЕСКИЙ АНАЛИЗ ЭФФЕКТИВНОСТИ АЛГОРИТМОВ PATCHWORK И КОХА ДЛЯ РЕАЛИЗАЦИИ СИСТЕМЫ ВОДЯНЫХ ЗНАКОВ**

*Ждан В.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Ярмолик В.Н. – д.т.н., профессор*

В работе рассматриваются угрозы интеллектуальной собственности, а также способы защиты от них. Особое внимание уделяется системам цифровых водяных знаков, в частности использованию стеганографических алгоритмов patchwork и коха в рамках их применения для реализации системы водяных знаков.

Для программного обеспечения и других объектов интеллектуальной собственности актуальными являются следующие угрозы: несанкционированное использование, обратное проектирование и несанкционированная модификация.

Одно из направлений для ликвидации данных угроз – системы цифровых водяных знаков. Данное направление использует различные методы: графическая стеганография, текстовая стеганография, обфускация и другие.

Алгоритм Patchwork является одним из базовых алгоритмов графической стеганографии. В его основе лежит статистический подход. Суть этого алгоритма состоит в следующем. При помощи

криптостойкого генератора псевдослучайных чисел, используя заранее известный секретный ключ, выбираются два пикселя изображения. Затем значение яркости одного из них увеличивается или уменьшается на некоторое значение в зависимости от типа встраиваемой информации. Значение яркости второго – уменьшается или увеличивается на ту же величину соответственно. Процесс повторяется около 10000 раз. Значение приращения варьируется от 1 до 5 [1].

Пусть значения яркостей пикселей, выбираемых на каждом шаге, равны  $a_i$  и  $b_i$ , а величина приращения –  $\delta$ . Тогда сумма разностей значений пикселей после  $n$  преобразований:

$$S_n = \sum_{i=1}^n [(a_i \pm \delta) - (b_i \mp \delta)] = \pm 2\delta n + \sum_{i=1}^n (a_i - b_i).$$

Математическое ожидание величины  $\sum_{i=1}^n (a_i - b_i)$ , то есть суммы разности значений пикселей в незаполненном контейнере равно нулю, и его значение в неизменном изображении будет много меньше чем  $n$ . Математическое ожидание величины  $S_n$  в заполненном контейнере будет равно  $\pm 2\delta n$ , а вычисленное значение  $S_n$  будет иметь большой соответствующий  $n$  порядок. В стегадекодере, используя ключ, проверяется значение  $S_n$ . Значению встроенного бита выбирается в зависимости от знака  $S_n$ . При положительной разности 1, а при отрицательной – 0 [2].

Таким образом, владелец может доказать свои интеллектуальные права, предъявив секретный ключ, который использовался для выборки изменяемых пикселей изображения.

В ходе реализации алгоритма Patchwork был разработан класс PatchworkInjector.

Данный класс реализует алгоритм Patchwork внедрения ЦВЗ в изображение. Для его использования пользователю необходимо ввести секретный ключ и строку, которая служит водяным знаком. Секретный ключ используется как «затравка» для генератора псевдослучайных чисел. Исходными данными служит путь к файлу, в который необходимо встроить водяной знак. Из данного файла считывается массив байт, представляющий собой цветные компоненты пикселей изображения. Каждый бит водяного знака встраивается в изображение, используя 15 000 пикселей, сгенерированных с помощью псевдослучайного генератора. Таким образом данный класс производит 7 500 Patchwork-итераций, на каждой из которых получает значение двух цветовых компонент изображения. Для первой из них увеличивает значение на  $\delta$ , для второй – уменьшает на  $\delta$ . Значение  $\delta$  выбрано равным 7. Изображение со встроенным водяным знаком отображается на экране.

Для проверки осуществляются те же действия, что и для встраивания. Вводится переменная, которая суммирует общее отклонение разности яркостей двух цветовых компонент. После 7 500 итераций в переменной остается отклонение, вызванное внедрением Patchwork-модификаций. Ошибка распределения принята равной 25 000. И если существует большее отклонение, то водяной знак считается подтвержденным.

Алгоритм Коха реализует внедрение ЦВЗ в области преобразования (служебную информацию) изображения. В качестве стегаконтейнера алгоритм использует коэффициенты дискретного косинусного преобразования (далее по тексту – ДКП), применяемого в широко распространенном формате JPEG.

ДКП производится отдельно для каждого из цветовых каналов. Исходное изображение разбивается на блоки размером  $8 \times 8$  пикселей. ДКП применяется по отдельности к каждому блоку, по которому вычисляются матрицы коэффициентов ДКП тем же размером  $8 \times 8$ . Коэффициенты обычно обозначаются через  $c_b(j, k)$ , где  $b$  – номер блока,  $(j, k)$  – позиция коэффициента внутри блока. Если блок обрабатывается в зигзагообразном порядке, как это имеет место в стандартном JPEG, то коэффициенты обозначаются через  $c_{b,j}$ . Коэффициент в левом верхнем углу  $c_b(0, 0)$  называется DC-коэффициентом и хранит информацию о средней яркости всего блока в целом. Остальные коэффициенты называются AC-коэффициентами и содержат информацию о характере распределения яркостей по блоку. В редких случаях может быть выполнено ДКП всего изображения целиком [3].

Алгоритм Коха предполагает внедрение одного бита ЦВЗ в блок размером  $8 \times 8$  пикселей. Выбирается значение порога встраивания  $\epsilon$ . Для передачи бита со значением 0 последовательным изменением двух выбранных псевдослучайно AC-коэффициентов, используя известный ключ, добиваются того, чтобы разность абсолютных значений коэффициентов была больше порога, а для передачи бита со значением 1 эта же разность делается меньше обратного значения порога [4]:

$$\begin{cases} |c_b(j_{i,j},k_{i,1})| - |c_b(j_{i,2},k_{i,2})| > \varepsilon, & s_i = 0, \\ |c_b(j_{i,j},k_{i,1})| - |c_b(j_{i,2},k_{i,2})| < -\varepsilon, & s_i = 1. \end{cases}$$

Для чтения ЦВЗ в декодере выполняется аналогичная процедура выбора коэффициентов по известному ключу, а значение переданного бита выбирается по следующей формуле [4]:

$$\begin{cases} s_i = 0, & |c_b(j_{i,j},k_{i,1})| > |c_b(j_{i,2},k_{i,2})|, \\ s_i = 1, & |c_b(j_{i,j},k_{i,1})| < |c_b(j_{i,2},k_{i,2})|. \end{cases}$$

В ходе использования алгоритмов Patchwork и Коха были получены следующие экспериментальные оценки, которые указаны в таблице 1:

Таблица 1 – Экспериментальные оценки использования алгоритмов Patchwork и Коха

№п/п	Алгоритм	Размер изображения	Количество байт водяного знака	Время встраивания водяного знака	Время извлечения водяного знака
1	Patchwork	1920x1080	4	18.9 с.	18.5 с.
2	Коха	1920x1080	4	25.3 с.	12.3 с.
3	Patchwork	1920x1080	8	37.8 с.	37.8 с.
4	Коха	1920x1080	8	25.6 с.	11.1 с.

В ходе сравнения эффективности алгоритмов удалось установить, что алгоритм Patchwork является достаточно стойким к операциям усечения, сжатия, изменения гистограммы изображения. Основным недостатком алгоритма – это неустойчивость к геометрическим преобразованиям: сдвигу, повороту, масштабированию. Другим недостатком является малая пропускная способность, что не дает возможности встраивания значительных по размеру цифровых водяных знаков.

Алгоритм Коха, в свою очередь, является легко реализуемым, достаточно простым и весьма эффективным. Однако до внедрения ЦВЗ нельзя оценить и повлиять на степень искажения изображения.

В ходе анализа экспериментальных оценок использования алгоритмов удалось установить, что при встраивании достаточно малых водяных знаков эффективнее алгоритм Patchwork. Алгоритм Коха в свою очередь одинаково эффективен при встраивании как малых, так и больших водяных знаков, следовательно, он является более универсальным.

**Список использованных источников:**

1. Семёнов К. П. Алгоритмы встраивания цифровых водяных знаков в растровые изображения / К. П. Семёнов, П. В. Зайцев // Информационная безопасность регионов : научно-практический журнал. – 2011. – №1. – С. 46–50.
2. Bender W. Techniques for Data Hiding / W. Bender, D. Gruhl, N. Morimoto, A. Lu // IBM Systems Journal. – 1996. – Vol. 35.
3. Демидчук А.И., Чернявский Ю.А. Алгоритм поиска в изображениях скрытых данных, встроенных методов Коха – Жао. Информатика. 2012;(1(33)):39-46.
4. Koch E. Towards Robust and Hidden Image Copyright Labeling / E. Koch, J. Zhao // IEEE Workshop on Nonlinear Signal and Image Processing. – 1995.

## РАСПРОСТРАНЕНИЕ МЕДИАСООБЩЕНИЙ В СЕТИ ИНТЕРНЕТ

*Козюкович А.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Куликов С.С. – канд. тех. наук*

Медиаинформация является основным типом информации в ресурсах сети Интернет. Она включает в себя фото-, видеоматериалы, а также обычный текст. Но авторам, как правило, необходимо поддерживать несколько ресурсов с одинаковым набором информации для охвата наибольшей аудитории. Для упрощения этого процесса можно прибегнуть к автоматизации.

В настоящее время в мире насчитывается более 4,1 млрд людей, которые имеют доступ к интернету, что составляет более половины населения Земли [1]. Все эти люди пользуются интернетом для разных целей: общения, работы, развлечения. Всё это связано с потреблением

различной информации. А находится такая информация на всяческих информационных ресурсах, например, в социальных сетях, на новостных порталах, в личных блогах и так далее.

Однако зачастую авторам приходится поддерживать множество ресурсов для публикации своих материалов. Это происходит в первую очередь с целью обеспечить наибольший охват аудитории: часто потребители информации могут использовать только определённый тип ресурса, как исходя из личных предпочтений, так и из локальной доступности [2]. Поэтому авторы стараются использовать как можно больше средств для распространения своих материалов и, как следствие, обеспечения наибольшего охвата аудитории и повышения своей популярности.

В настоящее время публикация материалов в большом количестве интернет-ресурсов приходится производить вручную: требуется и поддерживать учётные записи в нескольких социальных сетях, и публиковать статьи в личном блоге, и, возможно, ещё во многих ресурсах. Всё это занимает время. Кроме того, в крупных организациях на каждый из интернет-ресурсов может быть назначен отдельный редактор, что увеличивает расходы этой организации. Поэтому для упрощения этого процесса может быть разработано и использовано программное средство, автоматизирующее данный процесс.

Основой такого программного средства будут два понятия: медиainформация и интернет-ресурсы. Разберёмся с ними подробнее.

Программное средство должно поддерживать работу с различными видами информации. Наиболее распространёнными из них, которые и будут обязательными для программного средства, являются:

- текстовая информация;
- изображения;
- видеоролики;
- аудиозаписи;
- другие файлы.

Взаимодействие с интернет-ресурсами с помощью программного средства должно учитывать следующие особенности:

- требование авторизации на различных ресурсах в различных форматах;
- ограниченная поддержка видов информации ресурсами, на которых хочет публиковать материалы пользователь.

Кроме того, программное средство должно обеспечивать возможность отложенной публикации материалов в интернет-ресурсах.

Очевидно, что для наиболее эффективной работы этого программного средства поддержка лишь фиксированного набора интернет-ресурсов будет недостаточной - в сети существует огромное количество площадок, которое каждый день становится всё большим. Поэтому поддержка конкретных интернет-ресурсов будет достигаться поддержкой расширений, которые могут как поставляться с самим программным средством, так и добавляться пользователями.

Таким образом, идея разрабатываемого программного средства заключается в том, чтобы один раз собрать весь нужный набор медиаданных и использовать это программное средство для публикации материалов во всех требуемых интернет ресурсах сразу. Это обеспечит наиболее быструю доставку публикаций всем потребителям информации, что в конечном итоге может привести к росту охвата аудитории.

**Список использованных источников:**

1. Число пользователей интернета в мире выросло до 4,1 млрд человек - Общество - ТАСС // tass.ru [Электронный ресурс]. – Режим доступа: <https://tass.ru/obschestvo/7080150>.
2. Случаи запрета и блокировки социальных сетей в мире - РИА Новости, 16.05.2017 // ria.ru [Электронный ресурс]. – Режим доступа: <https://ria.ru/20170516/1494434193.html>

## **ИСПОЛЬЗОВАНИЕ АЛГОРИТМА ДЕРЕВА РЕШЕНИЙ ДЛЯ АНАЛИЗА МНОГОМЕРНЫХ ДАННЫХ НА ПРИМЕРЕ ДАННЫХ ПО ОНКОЛОГИЧЕСКИМ ЗАБОЛЕВАНИЯМ ЛЁГКИХ**

*Корховая А.Б.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Лапицкая Н.В. – зав. каф. ПОИТ*

В данной работе были рассмотрены и определены основные задачи для разработки алгоритмов по обработке медицинских данных онкологий лёгких. Были проанализированы подходы в машинном обучении и выбраны необходимые для данного решения задачи.



В контексте данной работы будут рассматриваться медицинские данные собранные на территории Беларуси по онкологическим заболеваниям лёгких. Вопрос проблемы лечения данного вида заболевания не теряет своей актуальности:

1) Рак является одной из основных причин смерти в мире: в 2018 году произошло 9,6 миллиона случаев смерти от рака;

2) Употребление табака является отдельным самым значительным фактором риска развития рака, который приводит к более чем 20% глобальных случаев смерти от рака и примерно 70% глобальных случаев смерти от рака легких;

3) По прогнозам исследований, число случаев заболевания от рака будет продолжать расти от 16 до 22 миллионов в следующие десятилетия.

С ходом развития технологий появились новые возможности использования знаний для поиска возможных путей предотвращения смертей от данного вида заболевания, например, основываясь на статистике построить линию протекания болезни больного в зависимости от различных факторов. Стоит отметить, что существуют успешные случаи излечения от онкологий лёгких, однако таких случаев очень мало.

Основной целью моего исследования стало определение пациента к одной из возможных категорий в зависимости от предложенных параметров (факторов).

Количество факторов, влияющих на исход заболевания, может быть огромнейшее количество и одной из технических задач становится определение значимых признаков, независимых друг от друга, для определения прогноза исхода заболевания [1].

Передо мной была поставлена задача: на основе предложенных данных по онкологии лёгких выявить подходящие алгоритмы анализа, которые в будущем врачи могли бы использовать для оценки возможных исходов лечения.

В предложенном датасете, для начала, было выделено 11 параметров, для которых будет проводиться анализ и в будущем прогнозироваться исход заболевания:

- 1) Лучевая терапия (1-была проведена, 0 – не проводилась).
- 2) Химия терапия (1 – была проведена, 0 – не проводилась).
- 3) Возраст (числовое значение).
- 4) Стадия рака лёгких, которая была установлена как диагноз (в диапазоне 1-4).
- 5) Шифр гистологии рака лёгких.
- 6) Жалобы (1 – пациент жаловался на плохое состояние, 0 – жалобы отсутствовали).
- 7) Пол пациента (1 – мужской, 0 - женский).
- 8) Онкоanamнез (1 – у родственников был рак, 0 – у родственников не встречалось данного заболевания).
- 9) Статус курения (1 – пациент курит, 0 – пациент не курит).
- 10) Операция (1 – проводилась, 0 – не проводилась).
- 11) Тип проводимой операции (зашифровано врачом: 0-3 – различные типы операций, 4 – операция не проводилась).

Параметр, определяющий результат заболевания, – зависимая переменная – также является бинарным: исход заболевания (1 – летальный исход; 0 – выздоровление).

Первым шагом в анализе данных стало определения их нормального распределения [1]. Результат оказался отрицательным: предложенные данные не имеют нормального распределения, а также очень мало количество выздоровевших в сравнении с умершими в результате болезни. График распределения данных представлен на рисунке 1:

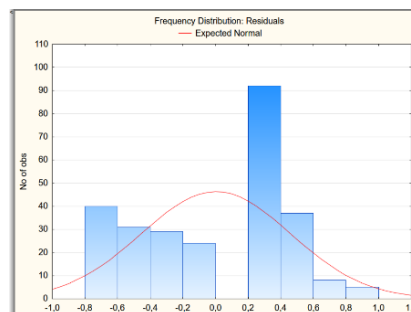


Рис. 1 – График нормального распределения данных

Основываясь на распределении данных, можно сделать вывод о нерациональности использования алгоритмов регрессионного анализа. Сама задача же заключена в определении определённого класса для пациента: выздоровеет или нет. Таким образом можно предположить о возможности применения дерева решений для решения поставленной задачи. Тип дерева решений определяется тем, что необходимо определить и так как зависящая переменная имеет дискретные значения, то деревья классификации станут наиболее подходящим решением [2].

При помощи ПО «STATISTICA» был проведен первичный эксперимент построения дерева решений (классификации) основываясь на модели, которую ПО генерирует сама и автоматически рассчитывает значимые признаки для данной модели. Полученные результаты представлены на рисунке 2 и 3:

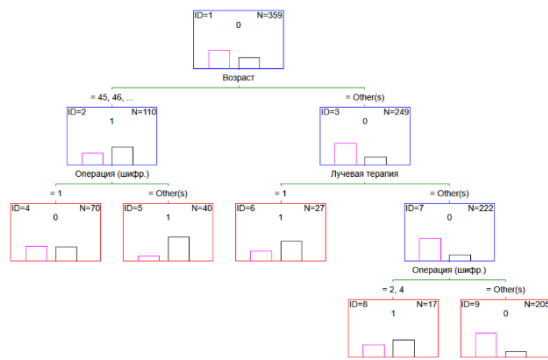


Рис. 2 – Дерево классификации

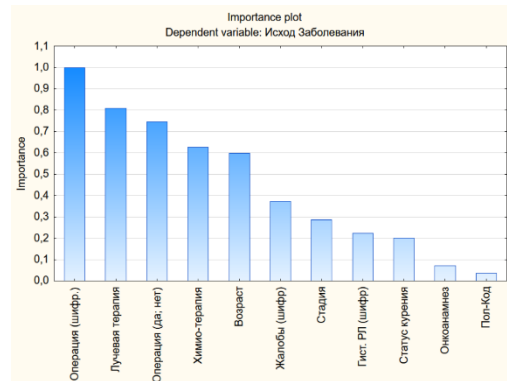


Рис. 3 – График значимости параметров выборки

Одна из сильных сторон дерева классификации является его гибкость, особенно важно для данной выборки его возможность работать с параметрами различных типов [3]: бинарных и числовых. Таким образом построенное дерево решение определила самые важные признаки, такие как возраст пациента, проводилась ли лучевая терапия и операция, и операция какого типа проводилась. Стоит отметить что самыми незначительными параметрами стали: пол, онкоанамнез и статус курения. Исходя из построенного алгоритма можно сказать, что данные параметры не должны влиять на протекание болезни онкологией лёгких.

В конечном счете, цель анализа с помощью деревьев классификации состоит в том, чтобы получить максимально точный прогноз. А максимально точный прогноз – это прогноз с минимальным числом неправильных классификаций [3]. Однако по полученным результатам, можно сказать, что для половины тестируемой выборки – классы распределяются неверно. И, следовательно, необходимы дальнейшие улучшения алгоритма, для решения данной задачи.

Данный этап моей работы на первоначальном этапе определил первые значащие признаки, которые влияют на протекание онкологии лёгких. Также было построено дерево решений, которое на основе данных признаков может отнести пациента к определённому классу.

Основываясь на построенных результатах можно сказать, что дерево решений не является оптимальным решением поставленной задачи, поэтому в дальнейшем моей целью станет применение алгоритма «Лес деревьев решений» и выбора оптимальных параметров для улучшения результатов алгоритма.

**Список использованных источников:**

- Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning. — Springer, 2001. ISBN 0-387-95284-5.
- Шахиди А. Деревья решений — общие принципы работы. URL: <http://www.basegroup.ru/library/analysis/tree/description/>
- Quinlan, J. R. Induction of Decision Trees // Machine Learning. Kluwer Academic Publishers. 1986. № 1. P. 81–106.

## РЕАЛИЗАЦИЯ ИНЕРЦИОННОСТИ ДИНАМИЧЕСКИХ ЭЛЕМЕНТОВ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА ПРОГРАММНЫХ СРЕДСТВ

*Кудрявцев М.Г.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Оношко Д.Е. – старший преподаватель*

Интерфейс программного средства зачастую не менее важен, чем его функционал. Неудобный и непонятный интерфейс может оттолкнуть потенциального пользователя, в то время как продуманный и логичный интерфейс способен повысить эффективность работы пользователя и сократить время знакомства с программным средством.

При реализации некоторых динамических элементов пользовательского интерфейса может оказаться полезным добиться визуального сходства с аналоговыми приборами. Например, в обычном термометре ртуть не может моментально достичь измеряемого уровня в силу определённых физических свойств и явлений. Такой же эффект можно использовать в программной реализации термометра, чтобы сохранить аутентичность. Однако полностью реализовывать физику таких

процессов, как термическое расширение вещества, не целесообразно. Гораздо проще и выгоднее вычислительно будет реализовать схожее поведение без использования физических формул.

Одно из возможных решений поставленной задачи было предложено в ходе разработки программного средства Спектроанализатор, который состоит из нескольких измерителей уровня в различных частотных полосах. Важным свойством спектроанализатора является баллистика – инерционность измерителей уровня. Она может регулироваться заданием скорости нарастания (атаки) и спада уровня [1]. Создание этого эффекта, базируется на двух принципах:

- новое значение уровня косвенно зависит от предыдущего;
- визуальное изменение уровня выполняется за несколько кадров.

Из-за того, что текущее положение измерителя зависит от его положения в предыдущий момент времени, исключаются резкие перепады. Теперь, чем дольше реальный уровень измеряемой величины держится без значительных изменений, тем ближе к нему становятся показания измерителя. Этот принцип позволяет уменьшить степень влияния резких кратковременных скачков уровня на показания измерителя, точно так же, как показания термометра практически не изменяются при кратковременном поднесении его к открытому пламени. Второй же принцип позволяет достичь плавности движения благодаря разбиению анимации на большее число кадров.

Реализация приведённого решения может выглядеть следующим образом: необходимо объявить три коэффициента, два из которых будут содержать вещественные значения в диапазоне от нуля до единицы – эти коэффициенты будут отвечать за скорости нарастания и спада показаний измерителя, а третья целочисленная величина будет равна количеству кадров, выделенному на отображение нового измерения.

Значения первых двух коэффициентов влияют на чувствительность измерителя следующим образом: чем ближе их значение к нулю, тем медленнее показания измерителя стремятся к реальному значению уровня, и наоборот. Например, если коэффициент на понижение уровня будет равен нулю, то измеритель никак не будет реагировать на падение величины и будет отображать только максимальное значение за данный промежуток времени.

Количество кадров на одно измерение может сильно варьироваться от диапазона возможных значений и от вычислительной сложности получения следующего измерения. Необходимо найти такое значение этого коэффициента, при котором бы сохранялись и плавность движения измерителя, и визуальное отсутствие задержки. Задержка возникает из-за того, что пока отображается одно измерение, следующее уже было получено. Важно, чтобы она была незаметна пользователю, ведь неотзывчивый интерфейс может сильно ухудшить общие впечатления о программном средстве.

Представленное решение может быть описано следующей формулой:

$$\Delta y = \frac{(y_{\text{нов.}} - y_{\text{пред.}})k_{\text{чувств.}}}{n} \quad (1),$$

где  $y_{\text{нов.}}$  - новое значение уровня;  $y_{\text{пред.}}$  – предыдущее значение уровня;  $k_{\text{чувств.}}$  – коэффициент чувствительности (к повышению уровня, если  $y_{\text{нов.}} > y_{\text{пред.}}$ , и наоборот);  $n$  – количество кадров.

Затем необходимо отобразить  $n$  кадров, каждый раз изменяя величину на  $\Delta y$ .

**Список использованных источников:**

1. Спектроанализатор – что мы на нем видим? [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://prosound.ixbt.com/education/spektr-analys.shtml>

## ОДНОПЛАТНЫЕ КОМПЬЮТЕРЫ В СИСТЕМАХ ПОЛУЧЕНИЯ ПОТОКОВЫХ АУДИОДАНЫХ

*Ермолович И.А., Пилинко Н.А., Фадеева Е.Е*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Леванцевич В.А. – м.т.н., старший преподаватель*

Описана реализация автономного приемника потоковых аудиоданных для воспроизведения передач с интернет радиостанций на базе одноплатного компьютера Raspberry Pi и операционной системы RaspbianOS.

В настоящее время широко распространено вещание радиостанций в УКВ диапазоне с весьма хорошим качеством. Однако у эфирных радиостанций есть несколько недостатков: во-первых, это неуверенный прием сигнала в некоторых районах, во-вторых ограниченный выбор радиостанций.

Целью проекта стало создание автономного приемника потоковых аудиоданных, способного воспроизводить вещание с интернет радиостанций on-line [1]. Для реализации проекта был выбран полярный микрокомпьютер Raspberry Pi, причем можно использовать любую модель: B, B+, 2 или Zero [2]. Подключение к сети Интернет может быть как проводным, так и по WiFi. В качестве операционной системы выбрана RaspbianOS как наиболее оптимизированная система для данного семейства микрокомпьютеров, позволяющая достичь максимальной производительности.

Система имеет клиент-серверную архитектуру. Сервер построен на связке LAMP (Linux, Apache, MySQL, PHP), клиент использует Python и Bash для управления и MPD (Music Player Daemon) для получения мультивещательного потока данных, а также имеет MPC (Media Player Classic) для упрощения управления MPD системы (Рис 1).

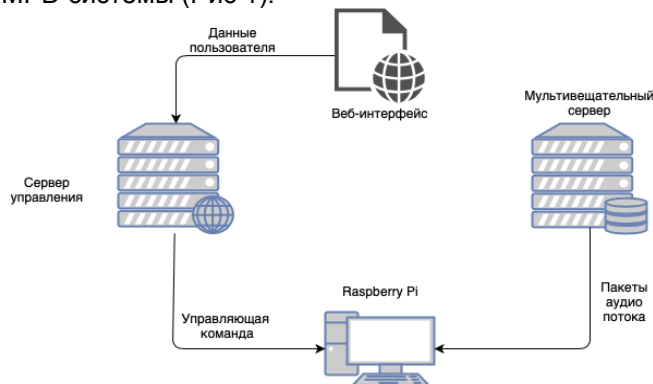


Рисунок 1 - Структура клиент-серверной системы

Программа на устройстве использует системные функции и не требует дополнительных библиотек для своей работы. Установка сделана на основе Bash-скрипта и самих файлов программы. По запуску Bash-скрипт помещает все необходимые файлы в загрузочные сектора системы и производит настройку программы для автозапуска при включении системы. Для работы графической части используется веб-страница, запускаемая в веб-браузере Chromium на устройстве. Данная связка позволяет производить улучшения системы без необходимости обновления ПО на самом устройстве. Также HTML+CSS предоставляет лучшие инструменты для проектирования пользовательского интерфейса, чем библиотеки поставляемые для Python. Программа на устройстве делает запрос на сервер управления каждые 300мс. Сервер сообщает устройству необходимые команды, которые оно впоследствии выполняет. Используется безопасный протокол HTTPS, поэтому дополнительного шифрования трафика не производится.

Список использованных источников

1. Как работает Интернет-радио <https://radio-tochka.com>
2. Пети, В. Микрокомпьютеры Raspberry Pi. Практическое руководство/ В.А. Петин – БХВ-Петербург, 2017.-240с

## РАЗРАБОТКА МОДЕЛИ КЛАССИФИКАЦИИ ПОЖАРНОЙ ОПАСНОСТИ В ЛЕСАХ

*Лабоха А.К., Шамына А.Ю.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Лапицкая Н.В. – канд. техн. наук, доцент*

В статье рассматриваются возможности совершенствования существующей системы классификации пожарной опасности в лесах Беларуси. Анализируется опыт других государств в данной области, возможность использования современных технических средств для прогнозирования потенциально опасных территорий. Исследуются технические аспекты использования численных моделей и данных дистанционного зондирования.

Лесные насаждения являются ценным ресурсом для любого государства. Однако их площадь и состояние могут ухудшаться вследствие действия различных факторов, и лесные пожары являются одной из основных причин. Согласно статистическим данным [1], за 2019-й год в Беларуси было зафиксировано 732 лесных пожара, а общая площадь, пройденная ими, составила 7371 га. С учетом изменений климатических условий и роста среднегодовой температуры на 0,5 градуса за последние 10 лет [2], задача предупреждения лесных пожаров приобретает особую актуальность.

При благоприятствующих условиях лесные пожары могут неконтролируемо распространяться, нанося существенный вред природным экосистемам и поселениям человека, поэтому важной задачей

является снижение вероятности возникновения пожара. Для этого необходимо отслеживать текущий уровень пожарной опасности в лесах, чтобы грамотно перераспределять ресурсы для проведения профилактических мероприятий.

Согласно действующему в Республике Беларусь стандарту СТБ, класс пожарной опасности в лесу и показатель горимости леса по условиям погоды рассчитывается по комплексному показателю Н.А. Диченкова. При расчете данного показателя используются следующие данные:

- температура воздуха (в градусах) в 14 часов;
- точка росы;
- число сухих суток.

При этом учет множества факторов, которые могут оказать влияние на распространение пожара, таких как структура лесного фонда, грозовая активность, влажность подстилающей поверхности, в стандарте не описан. Также следует принимать во внимание, что причиной большинства природных пожаров является человек.

Еще одним важным аспектом является пространственная точность. В настоящий момент источниками метеорологических сведений при определении пожарной опасности являются метеостанции, которые распределены по территории страны неравномерно и в большинстве своем находятся в непосредственной близости от крупных населенных пунктов [3]. По этой причине показатель горимости не может предоставить достоверных и исчерпывающих сведений о состоянии лесного фонда, в особенности труднодоступных территорий. При этом проблема точности метеорологических данных может быть решена путем использования численных моделей прогнозирования погоды. Данные прогностической модели WRF, предоставляемые Белгидрометом, имеют пространственное разрешение 15 км и временной шаг 1 час [4].

Методика определения пожарной опасности в лесах была изменена в Польше в 2015-м году [5]. В принятых правилах выделяется понятие «прогнозируемый уровень риска лесных пожаров» наряду с термином «уровень риска лесных пожаров». Согласно методике, степень риска лесных пожаров определяется на основании измерений:

- влажности подстилки в сосновых насаждениях 3-го класса в молодом хвойном лесу или смешанном хвойном лесу;
- температура воздуха, измеренная на высоте 0,5 м от заболоченного участка около станции измерений;
- относительной влажности воздуха, измеренной на высоте 0,5 м от дерновой зоны около станции измерений;
- количество осадков за последние сутки.

Также в методике регламентируется количество станций измерения показателей на площадь в зависимости от пожарной категории лесов. Очевидно, что принятый в Польше подход неприменим в текущих условиях для Беларуси, поскольку, помимо необходимости апробации подхода на тестовом участке, потребуются значительные мгновенные инвестиции в размещение станций сбора данных.

В 2009-2011 годах была разработана новая система ежедневного прогнозирования опасности лесных пожаров (FFDFS), использующая данные дистанционного зондирования, и внедрена в северной части канадской провинции Альберта в периоды пожаров 2009–2011 годов [6]. Авторами была поставлена задача улучшить временное разрешение существовавшей модели с восьмидневной до ежедневной. Созданная модель рассчитывала спектральные вегетационные индексы, такие как нормализованный разностный вегетационный индекс NDVI, нормализованный разностный индекс воды NDWI и другие, а также поверхностную температуру  $T_s$  на основе материалов космической съемки, накопленных за 8 дней. Также при моделировании использовалось суточное накопленное количество воды как суррогат переменных, связанных с осадками и влажностью.

Построенная модель была верифицирована с помощью данных о фактических пожарах в период 2009-2011 годов. Модель, учитывающая все переменные, в том числе вегетационные индексы, продемонстрировала отличные результаты: в среднем 95,51% пожаров попадали в классы пожарной опасности от «чрезвычайно высоких» до «умеренных». При этом следует отметить, что использование только отдельных вегетационных индексов приводило к точности классификации порядка 50%; таким образом, для повышения точности прогнозирования необходимо рассматривать показатели вегетационных индексов в совокупности.

Результаты исследований канадских ученых в построении системы классификации пожарной опасности с использованием данных дистанционного зондирования могут быть использованы и для территории Беларуси. Однако необходимо уделить особое внимание сложностям, которые могут оказать значительное влияние на точность прогнозирования.

При использовании аэрокосмических снимков существенной проблемой является облачность при выполнении снимков, которая не позволяет получить актуальные сведения об исследуемой территории. Так, для территории Воложинского района только 16 аэрокосмических снимков из 72-х, выполненные летом 2019 года спутником системы Sentinel-2, имели степень облачности менее 20% (для получения информации о всей территории данного района необходимы 4 снимка) [7]. Поэтому

важно использовать данные дистанционного зондирования различных систем, поскольку выполнение снимков в различные дни повышает вероятность получения четкого изображения местности.

В работе канадских исследователей также был использован алгоритм заполнения отсутствующих данных [8], чтобы определить и заменить загрязненные облаками пиксели изображения. Для замены загрязненных пикселей использовались как соседние пиксели в пространстве, так и значения этого же пикселя в соседние моменты времени. Реализация подобного алгоритма не позволила полностью устранить влияние облачности во время аэрокосмической съемки, однако повысила процент данных, которые можно использовать в моделировании, в результате исправления от 84,7% до 98,9% загрязнённых пикселей.

При построении системы классификации пожарной опасности для территории Беларуси также могут быть использованы накопленные статистические данные о фактических пожарах. Поскольку чаще всего лесные пожары возникают по вине человека [9], в результате анализа статистических данных могут быть определены исторически сложившиеся зоны повышенной пожарной опасности. Также при определении мест наибольшего влияния человека могут быть использованы сведения о населенных пунктах и плотности населения, дорогах и местах отдыха.

Верификацию полученной модели следует проводить по фактическим данным о лесных пожарах на территории Беларуси в 2020-м и последующих годах в пожароопасный период. Ввиду особенностей прогнозирования и отсутствием стандартизации вегетационных индексов, следует разработать несколько тестовых моделей с различными наборами вегетационных индексов и провести сравнительную характеристику их эффективности. С учетом вышеизложенного, при построении модели классификации пожарной опасности в лесах также важно осуществлять тесное взаимодействия со специалистами из сферы лесного хозяйства, которые могут провести компетентную экспертную оценку свойств модели и полученных результатов на основе накопленного практического опыта работы.

**Список использованных источников:**

1. Годовые данные. Лесное хозяйство [Электронный ресурс]. – 2020. – Режим доступа: <https://www.belstat.gov.by/ofitsialnaya-statistika/realny-sector-ekonomiki/selskoe-hozyaistvo/lesnoe-khozyaystvo/godovye-dannye/> – Дата доступа: 03.04.2020.
2. Среднегодовая температура воздуха в Беларуси за 10 лет увеличилась на 0,5 градуса [Электронный ресурс]. – 2020. – Режим доступа: <https://www.belta.by/society/view/srednegodovaja-temperatura-vozduha-v-belarusi-za-10-let-uvlechilas-na-05-gradusa-377805-2020/> – Дата доступа: 03.04.2020.
3. BelMeteo.Net - Метеостанции Беларуси, метеорологические станции Республики Беларусь [Электронный ресурс]. – 2020. – Режим доступа: <http://belmeteo.net/stations.html> – Дата доступа: 05.04.2020.
4. POGODA.BY | О сайте | Белгидромет [Электронный ресурс]. – 2020. – Режим доступа: <http://www.pogoda.by/about/?information=forecast> – Дата доступа: 05.04.2020.
5. Rozporządzenie Ministra Środowiska z dnia 13 lipca 2015 r. zmieniające rozporządzenie w sprawie szczegółowych zasad zabezpieczenia przeciwpożarowego lasów [Электронный ресурс]. – 2015. – Режим доступа: <http://prawo.sejm.gov.pl/isap.nsf/DocDetails.xsp?id=WDU20150001070> – Дата доступа: 07.04.2020.
6. Chowdhury, E.H.; Hassan, Q.K. Development of a New Daily-Scale Forest Fire Danger Forecasting System Using Remote Sensing Data. Remote Sens. 2015, 7, 2431-2448.
7. Copernicus OAH [Electronic resource] – 2020. – Access mode: <https://scihub.copernicus.eu/dhus/>. – Access date: 2020/04/09.
8. Chowdhury, E.H.; Hassan, Q.K. Use of remote sensing-derived variables in developing a forest fire danger forecasting system. Nat. Hazards 2013, 67, 321–334.
9. От пожаров | Официальный сайт Министерства лесного хозяйства Республики Беларусь [Электронный ресурс]. – 2020. – Режим доступа: <https://www.mlh.by/our-main-activites/safety-and-security/ot-pozharov/> – Дата доступа: 09.04.2020.

## АЛГОРИТМЫ ЭМИССИИ БАНКОВСКИХ КАРТ

*Ламчановский А.Г.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Бахтизин В.В. – к.т.н., профессор*

В докладе рассматривается вопрос автоматизации заказов банковских карт, построены алгоритмы эмиссии банковских карт.

Эмиссия (выпуск) и обслуживание банковских карт является ключевым процессом банка в сфере обслуживания клиентов. Подразделением банка, отвечающим за работу карточных процессов, является процессинговый центр (ПЦ) – структурное подразделение банка или внешняя организация, осуществляющее информационное и технологическое взаимодействие между участниками расчетов и обеспечивающее проведение внутрибанковской обработки операций с банковскими картами. Основным программным обеспечением ПЦ является карт-система. Для заказа карт разработана система учета заявок на выпуск карт, представляющая собой отдельное ПО, взаимодействующее с карт-системой, SMS-системой банка и мобильным банковским приложением.

В докладе рассмотрены алгоритмы эмиссии карты для банка, пользующегося услугами внешней процессинговой компании (алгоритм 1), и для банка с внутренним процессинговым центром (алгоритм 2). Алгоритм 1 состоит из следующих шагов.

Шаг 1: получить от клиента письменное заявление или телефонный звонок с просьбой выпустить платежную карту. Перейти к шагу 2.

Шаг 2: сотрудник отделения регистрирует обращение клиента в системе учета заявок. Перейти к шагу 3.

Шаг 3: сотрудник департамента процессинга формирует реестр на выпуск карт и передает его внешней процессинговой компании. Перейти к шагу 4.

Шаг 4: внешняя процессинговая компания изготавливает карты и передает их в департамент процессинга банка заказчика. Перейти к шагу 5.

Шаг 5: департамент процессинга сортирует карты по отделениям и передает их курьерам специальной связи, доставляющим карты в целевые отделения. Перейти к шагу 6.

Шаг 6: целевое отделение получает карты и информирует клиента о возможности получить карту посредством SMS-сообщения. Если клиент приходит для получения карты в течение 90 дней, перейти к шагу 7. Иначе перейти к шагу 8.

Шаг 7: сотрудник отделения выдает карту клиенту. Завершить алгоритм.

Шаг 8: карта уничтожается. Завершить алгоритм.

Данный алгоритм выгоден для небольших банков, не имеющих возможности содержать собственный процессинговый центр. Кроме того, данный алгоритм позволяет выпускать клиентам срочные карты, оперативно доставляемые курьерами спецсвязи в отделения.

Переход банка на собственный процессинговый центр позволяет сократить расходы на обслуживание карт, сократить время на настройку новых карточных продуктов и ускорить взаимодействие за счет более глубокого доступа к настройкам процессингового центра и отсутствия необходимости подстраиваться под промежуточное звено в виде банка поставщика услуг. Это позволяет предоставлять более качественный сервис и вводить новые продукты раньше конкурентов. Для этого в систему учета заявок добавлена интеграция с почтовой системой, что позволяет клиентам отслеживать этапы доставки конверта через мобильный банк.

Алгоритм 2 состоит из следующих шагов:

Шаг 1: клиент в банковском мобильном приложении формирует заявку на выпуск карты, указывает желаемый продукт, тип карты и адрес для доставки. Перейти к шагу 2.

Шаг 2: система учета заявок передает заявки в карт-систему. Перейти к шагу 3.

Шаг 3: сотрудники персособюро (подразделения банка, отвечающего за нанесение на заготовки пластиковых карт их реквизитов) департамента процессинга печатают карты. Перейти к шагу 4.

Шаг 4: персособюро передает карты курьеру почтовой службы. Если почтовый ящик по указанному адресу существует, перейти к шагу 5. Иначе перейти к шагу 6.

Шаг 5: карты доставляются клиенту в почтовый ящик. Перейти к шагу 7.

Шаг 6: почтовая служба возвращает карты в банк для уничтожения. Завершить алгоритм.

Шаг 7: клиент забирает конверт с картой из почтового ящика и активирует её в мобильном банковском приложении. Завершить алгоритм.

Помимо сокращения накладных расходов на производство и обслуживание карт, алгоритм позволяет снять нагрузку с отделений банка, специалистам не нужно заказывать и выдавать карты клиентам. Клиентам не нужно посещать отделения банков, карты будут доставлены сотрудниками почты к ним домой. Минусом алгоритма является то, что он не позволяет изготавливать карты клиенту день в день, так как почтовые службы выполняют доставку в течение семи дней.

## **ЗАГРУЗКА И СОВМЕСТИМОСТЬ ОПЕРАЦИОННЫХ СИСТЕМ**

*Ласый А.М.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Оношко Д.Е. – старший преподаватель*

Одним из основных требований к современной операционной системе является возможность запуска и нормальной работы приложения, скомпилированного или для более старой версии данной, или для совершенно другой операционной системы. В данной статье на примере простейшей операционной системы рассмотрены основные моменты, связанные с ее загрузкой и достижением бинарной совместимости.

Проблема бинарной совместимости операционных систем на данный момент является одной из самых актуальных. Каждый день пишется огромное количество новых программ. Многие из этих программ пользователи будут запускать в будущем еще достаточно долго. Однако с течением

времени поддерживать старые программы становится все сложнее, так как операционные системы развиваются достаточно быстро.

В то же время стоит проблема загрузки операционных систем с таких устройств, как флеш-накопитель, в CSM BIOS или в Legacy BIOS, поскольку предоставляемая по этому поводу информация в большинстве своем разрозненна и не согласованна.

Для изучения данных вопросов была разработана с нуля небольшая операционная система для реального режима x86-совместимых процессоров.

Разработка велась на языке ассемблера, что дало возможность максимально контролировать всю происходящую работу с данными. Размер ядра операционной системы составил порядка одного килобайта.

В качестве пользовательского интерфейса с помощью прерывания BIOS 10h была реализована графическая оболочка, позволяющая отображать имеющиеся на диске файлы и запускать их.

Так как самой известной операционной системой, которая работает в реальном режиме, является MS-DOS, целью разработки новой операционной системы с точки зрения бинарной совместимости являлась поддержка COM-файлов. COM-файлы представляют собой один из простых типов исполняемых файлов. Они широко использовались в 70-80-е годы 20 века в операционных системах ввиду своей простоты и удобства, так как код, стек и данные располагались в одном сегменте. [1]

Для достижения поддержки данного типа файлов необходимо было обеспечить корректные запуск и завершение программы. Первое достигается за счёт установки выполнения выбранной программы со смещения 100h в сегменте. Второе же требует реализации прерывания DOS 20h, а также минимального заполнения префикса программного сегмента. В частности, особую важность представляют два байта префикса, поскольку именно там в COM-файлах располагается инструкция INT 20h. Доступ к этой инструкции можно получить посредством предварительной записи операционной системой в стек числа 0 с его последующим считыванием инструкцией ret.[1]

В результате поддержка COM-файлов была успешно осуществлена. Тестирование проводилось как на собственных программах, так и на старых видеоиграх, которые были скомпилированы в 1980-х годах. В качестве примера можно привести видеоигры Pac-Man и Paratrooper. Все тестируемые программы запускались и работали без ошибок.

В качестве файловой системы для данной операционной системы была выбрана FAT12, так как она применяется на дискетах и схожа с файловой системой FAT32, которая до сих пор используется на флеш-накопителях, а также проста и совместима со многими операционными системами.[2]

Для реализации считывания файлов из области данных FAT12 было реализовано два обработчика программных прерывания. Первое осуществляет считывание с накопителя в определенный буфер заданное количество секторов. Второе прерывание, используя первое загружает в буфер файл с указанным именем.

Операционная система была записана на дискету и на флеш-накопитель. Целью опыта являлось успешно запустить ее на разных устройствах. Для записи на накопители был сформирован образ загрузочной дискеты размером 1.44 Мб. Запись образа на дискету с последующим запуском операционной системы с нее трудностей не вызвал, тогда как запись на флеш-накопитель была проблематична.

На данный момент существует три варианта загрузки операционной системы с флеш-накопителя:

- Как единственный FAT-12 раздел.
- Как Hard Disk Drive с несколькими разделами.
- Как оптический диск.

В данном эксперименте был выбран второй вариант, поскольку он максимально аналогичен запуску с реального HDD. Помимо этого на языке ассемблера проще генерировать FAT-образ, чем образ оптического диска. Однако и этот вариант имеет ряд своих затруднений. Основной проблемой являлось то, что утилиты, осуществляющие разделение диска на тома, при работе с одним и тем же диском получают совершенно разные результаты.

Как следствие, правильно записать значение BPB\_HiddSec в загрузочном секторе становится сложно. При этом в случае, если значение окажется неверным, загрузочный сектор будет считывать не с того места, которое нас интересует.

Таким образом, необходимо вручную определять смещение раздела и на основе полученного результата высчитывать количество секторов, на которое был смещен загрузочный сектор. В данном эксперименте смещение составило 1 Мб. Размер сектора насчитывает 512 байт. Имея все необходимые данные, смещение в пересчете на количество секторов можно рассчитать по следующей формуле:

$$K = \frac{0}{S} \quad (1),$$



где  $K$  – смещение загрузочного сектора в пересчете на секторы;  $O$  – смещение загрузочного сектора;  $S$  – размер сектора в байтах.

Получившаяся в данном эксперименте с помощью утилиты diskpart структура разделов флеш-накопителя представлена на рисунке 1:

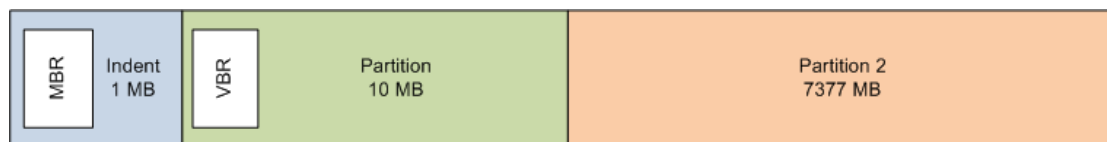


Рисунок 1 – Структура разделов флеш-накопителя

В самом начале диска размещается MBR (master boot record), задача кода которого заключается в нахождении и переходе в нужный раздел диска. А в начале загрузочного раздела с операционной системой расположен VBR (volume boot record), который содержит код по поиску, а также дальнейшей загрузке операционной системы (в данном эксперименте происходила загрузка уже непосредственно ядра, однако во многих случаях загружается так называемый вторичный загрузчик, который и выполняет загрузку ядра). Кроме того, стоит отметить, что не упомянутый ранее раздел 2, представленный на рисунке 1, можно отформатировать и использовать как обычный флеш-накопитель.

По итогу после создания структуры разделов и подсчета смещения  $O$  величина  $K$  согласно формуле (1) составила 2048 секторов.

Помимо этого, запись образа на выбранный накопитель может быть затруднена из-за немногочисленности программного обеспечения, предоставляющего удобный интерфейс для записи отдельных секторов и корректно работающего в современных версиях операционной системы Windows.

Также стоит отметить, что после осуществления записи необходимо вручную сделать раздел, на который записана операционная система, активным.[3] В противном случае загрузка с данного раздела происходить не будет.

Для тестирования операционной системы было отобрано несколько моделей персональных компьютеров:

- Dell Inspiron 15 7000.
- ASUS Celeron - M540.
- ASUS K53T.
- HP 250 G5

Отбор проводился с целью охвата как можно большего диапазона технических характеристик персональных компьютеров. На всех моделях операционная система загружалась корректно и функционировала в полной мере.

**Список использованных источников:**

1. Электронный справочник TECH Help! 6 by Flambeaux Software
2. Microsoft Extensible Firmware Initiative FAT32 File System Specification
3. DiskPart Documentation on docs.microsoft.com/en-us/windows-server/administration/windows-commands/diskpart

## **ПРОГРАММНОЕ СРЕДСТВО ДЛЯ ХРАНЕНИЯ И АНАЛИЗА РЕЗУЛЬТАТОВ СОЦИОЛОГИЧЕСКИХ ИССЛЕДОВАНИЙ С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ ASP.NET**

*Лащиновская К.Д.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Петюкевич Н.С. – ст. преподаватель*

В статье дано краткое описание функциональности программного средства для хранения и анализа результатов социологических исследований. Также приводится обоснование некоторых архитектурных решений, принятых при проектировании программного продукта.

Социологическое исследование представляет собой деятельность, направленную на получение объективных знаний о социальных отношениях, явлениях и процессах. Социологические исследования в зависимости от задач, которые они решают, и методов, которыми они пользуются,

могут быть подразделены на теоретические и эмпирические. Теоретические исследования ставят целью развитие и совершенствование научных представлений об изучаемом предмете, а эмпирические (прикладные) посвящены решению конкретной социальной проблемы. Результаты эмпирических социологических исследований необходимы для более эффективного управления предприятием, отраслью, страной. На сегодняшний день социологический анализ активно применяется для построения рекламных кампаний, оценки мнений сотрудников организаций и учащихся учебных заведений. Одним из наиболее удобных методов сбора информации для социологического исследования является опрос. Анализ данных, полученных в результате проведения опроса, позволяет сформулировать утверждения, характеризующие испытуемую группу лиц.

Для анализа результатов эмпирических социологических исследований часто применяются такие программные средства, как SPSS, Stata и SurveyMonkey. Достоинством SPSS и Stata является очень мощный аналитический инструментарий. Эти программные продукты имеют встроенный командный язык для получения нужных выборок данных и их анализа. Однако при использовании данных программных средств результаты проведения опросов и отчеты хранятся на локальном компьютере, что делает затруднительной работу в команде исследователей. Веб-сервис SurveyMonkey является более современным решением, которое позволяет совместно создавать опросы и анализировать полученные результаты в online-режиме. Однако при проведении опросов в нескольких странах (на нескольких языках) данные, полученные из разных стран, объединяются в одну выборку [1]. Таким образом, отсутствует возможность проводить аналитику в разрезе стран.

В разработанном программном средстве были учтены недостатки аналогов. Данное программное средство представляет собой веб-приложение, которое позволяет работать командам исследователей, а также поддерживает работу с исследованиями, проводимыми в нескольких странах. Приложение ориентировано на мелкие и средние компании, специализирующиеся на проведении эмпирических социологических исследований, а также на организации, которые периодически проводят социологические опросы среди своих сотрудников.

При создании нового исследования существует возможность выбрать пользователей, которые будут иметь доступ к нему. Для каждого пользователя устанавливается уровень доступа к исследованию: наблюдение (только просмотр результатов анализа), работа с данными (загрузка новых данных исследования), аналитика (построение диаграмм и отчетов) и полный доступ к исследованию. Указание уровней доступа позволяет разделить обязанности в команде исследователей.

Для создания исследования необходимо указать опорные вопросы и варианты ответа на них. Доступны следующие виды вопросов: вопросы с единственным выбором, вопросы с множественным выбором, открытые вопросы с текстовым ответом. Если исследование проводится на нескольких языках, для каждого вопроса и вариантов ответа на него нужно указать соответствие для выбранного языка.

Добавление данных в приложение осуществляется путем загрузки файла в формате XLSX, содержащего ответы респондентов на вопросы исследования. Для корректного считывания данные в файле должны быть организованы по единому шаблону. За шаблон файла данных был взят файл, который генерируется сервисом для создания опросов Google Forms. При таком форматировании файла заголовками колонок таблицы являются вопросы исследования на одном из выбранных языков, а каждая строка таблицы содержит ответы одного респондента.

Разработанное приложение предоставляет достаточно широкие аналитические возможности.

Для вопросов с вариантами ответа доступен частотный анализ для каждого варианта ответа. Результат анализа может быть представлен в виде круговой или столбчатой диаграммы, при этом анализ можно проводить в разрезе стран и временных промежутков, в которых проводилось исследование. Также существует возможность строить диаграммы корреляции, для чего выбирается независимый вопрос и вопрос, зависящий от него. Для каждого варианта ответа независимого вопроса строится диаграмма, отражающая распределение по вариантам ответа для зависимого вопроса. Все диаграммы, построенные в программном средстве, могут быть сохранены на компьютер в виде изображения в формате PNG.

Для вопросов с выбором одного варианта ответа существует возможность вычисления коэффициента корреляции. Коэффициент корреляции – это мера оценки статистической взаимосвязи между парой (как правило) или большим количеством переменных. Нужный коэффициент выбирается исходя из типа шкалы переменной (номинальная, порядковая, интервальная, метрическая шкалы). В большинстве случаев каждый вариант ответа на вопрос имеет свой идентификатор, который не зависит от эмпирического значения варианта ответа. К вопросам, имеющим подобные варианты ответа, относятся, например, вопросы по гендерной принадлежности испытуемого, его национальности, семейном положении, роде деятельности. В этом случае переменные, хранящие результаты ответов на вопросы, относятся к номинальной шкале. Для переменных номинальной шкалы может быть получен коэффициент корреляции Крамера [2]. Данный

коэффициент измеряется в пределах от 0 до 1. Чем больше значение коэффициента, тем сильнее связь между переменными.

Для вопросов, предполагающих текстовый ответ, существует возможность создания диаграммы «облако слов». Для построения диаграммы все текстовые ответы анализируются по частоте встречаемости каждого слова. Найденные слова размещаются на диаграмме, причем размер шрифта каждого слова пропорционален частоте встречаемости слова среди всех ответов.

В программном средстве также существует возможность создавать фильтры для выборки данных. Условия фильтра могут быть объединены посредством оператора И либо ИЛИ. Данные, подходящие под фильтр, могут быть загружены на локальный компьютер в формате XLSX.

При проектировании программного средства большое внимание было уделено архитектуре хранилища данных. База данных средства анализа результатов опроса должна поддерживать возможность работы с большими объемами данных, а также должна позволять быстро извлекать ответы респондентов по переданным условиям.

В разработанном приложении для удобства выборки данных таблицы в базе данных построены по архитектуре «звезда»: таблица, содержащая ответы респондентов, является центральной и связана через внешние ключи с таблицами вопросов, вариантов ответов, респондентов и стран. Необходимо отметить, что подобная схема связи таблиц является основой для создания многомерной базы данных (куба) посредством использования службы Microsoft Analysis Services. Многомерная база данных является более совершенным решением для построения аналитических систем, чем реляционная база данных, так как в кубе данные заранее агрегируются по различным измерениям, и существует возможность делать множественные «срезы» данных по нужным параметрам. Однако весь аналитический функционал в разработанном приложении может быть реализован через вызов хранимых процедур и использование куба является избыточным. Поэтому было принято решение отказаться от многомерной базы в пользу обычной реляционной базы.

Так как данное программное средство призвано решить, в том числе, проблему работы над исследованием в команде, результаты проведения социологических опросов должны храниться на компьютере, к которому имеют доступ все исследователи в команде. При этом пользователи должны иметь возможность работать с данными независимо друг от друга. Таким образом, было принято решение о построении приложения по клиент-серверной архитектуре с использованием принципов REST. Серверная часть программного средства была написана при помощи технологии ASP.NET WebApi, которая позволяет создавать RESTful-сервисы. Для работы с файлами в формате XLSX была использована библиотека EPPlus. Преимуществом данной библиотеки является то, что она не имеет сторонних зависимостей и может функционировать без установленного на сервере Microsoft Excel. В качестве базы данных была выбрана MSSQL Server 2019. Для доступа к данным были использованы EntityFramework (с подходом database-first), а также библиотека ADO.NET (использовалась для вызова хранимых процедур). Клиентская часть программного средства была написана на основе JavaScript-фреймворка Vue, характеризующегося высокой скоростью работы и наличием большого количества устанавливаемых компонентов.

**Список использованных источников:**

1. Документация для сервиса SurveyMonkey [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://help.surveymonkey.com>
2. Статистики, рассчитываемые для таблиц сопряженности [Электронный ресурс]. – Электронные данные. – Режим доступа: [https://www.ibm.com/support/knowledgecenter/ru/SSLVMB\\_25.0.0/spss/base/idh\\_xtab\\_statistics.html](https://www.ibm.com/support/knowledgecenter/ru/SSLVMB_25.0.0/spss/base/idh_xtab_statistics.html)

## **ВРМ СИСТЕМЫ КАК ФАКТОР ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ПРЕДПРИЯТИЯ**

*Манцевич Андрей Валерьевич*

*Белорусский Государственный Университет Информатики и Радиоэлектроники, кафедра менеджмента, г. Минск, Республика Беларусь*

*Домакур О.В. – канд. эконом. Наук*

Врм системы, необходимость применения врм систем в структуре предприятия, практическая эффективность на примере предприятия.

В текущее время эффективность любого предприятия зависит не только от его производственной мощности, квалификации сотрудников, обеспеченности в финансировании и инвестициях, но также и в правильном и рассчитанном менеджменте, контроле и управлении всеми его структурными подразделениями.

Любому предприятию необходимо развиваться в условиях современной экономики, когда конкуренция постоянно растет, а количество сторонних предложений растет. Однако необдуманно подходить к данному процессу невозможно, необходимо четкое планирование и моделирование всех возможных путей модификации предприятия. Специально для этого и создаются bpm системы.

BPM системы – это класс программных продуктов, которые помогают управлять бизнес процессами организации. Данное программное обеспечение позволяет смоделировать каждый отдельный процесс в компании, представить его в графической схеме. Также позволит управление, анализ и дальнейшее совершенствование бизнес процессов путем моделирования возможных изменений в процессе в зависимости от основных критериев.

BPM системы предназначены для повышением эффективности работы организации, путем реорганизации процессов и улучшения системы управления. Этот класс программного обеспечения создавался с таким расчетом, чтобы можно было быстро определить слабые места процессов и подразделения компании, которые снижают общую эффективность работы, проанализировать их, и выдать расчетные модели по возможности улучшения ситуации.

Среди всех специализированных информационных систем, позволяющих анализировать, моделировать, управлять и отображать бизнес-процессы предприятия, BPM системы находятся на стыке трех сфер корпоративного программного обеспечения (рис. 1).



Рисунок 1 – Сочетание классов ПО в BPM системе

После внедрения bpm систем в структуру управления предприятием, отмечалось увеличение гибкости в области принятия решений, уменьшение нагрузки на отделы анализа и планирования.

При проведении математического расчета при моделировании коэффициент ускорения принятия решений на основе работы системы составляет 1.63x от стандартного. Хотя данные были собраны автором только на одном предприятии, следует предположить, что система может эффективно показать себя и на других предприятиях.

#### **Вывод**

Необходимо привлечение bpm систем с структуру управления предприятия и увеличение общего уровня заинтересованности специалистов в работе с данными системами. Этого можно добиться субсидированием в сферу образования на уровне университетов, путем предоставления учащимся возможности изучать материалы и пользоваться существующими платформами для создания bpm систем. А также использованием bpm систем на различных предприятиях, в качестве примера эффективности данных систем.

#### **Список использованных источников:**

1. Pega Platform Foundation Exercise Guide / Pegasystems Inc. One Rogers Street, 2010.
2. System Architect Essentials Exercise Guide/ One Rogers Street Cambridge, MA 02142-1209, 2009.

## **ТЕНДЕНЦИИ РАЗВИТИЯ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ В БГУИР**

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Марков А.Н., Божков Р.А.*

*Парамонов А.И. – к.т.н., доцент*

За последние 20-30 лет в современном мире потребность в вычислительных мощностях росла в геометрической прогрессии как для обычных пользователей, так и для использования в промышленных масштабах. В связи с этим, как и предполагается, её ресурсы вычислительных мощностей внушительно возросли. Как следствие, учитывая текущий уровень развития, вырос и

объём используемых данных.

По данным исследовательской компании IDC (International Data Corporation) представлен график роста количества цифровой (структурированной и неструктурированной) информации в мире за 2020 год.

**Структурированные данные** – данные, находящиеся в реляционных базах и электронных таблицах.

**Неструктурированная информация** – это информация, которая либо не имеет заранее определенной структуры данных, либо не организована в установленном порядке. График представлен на рисунке 1:

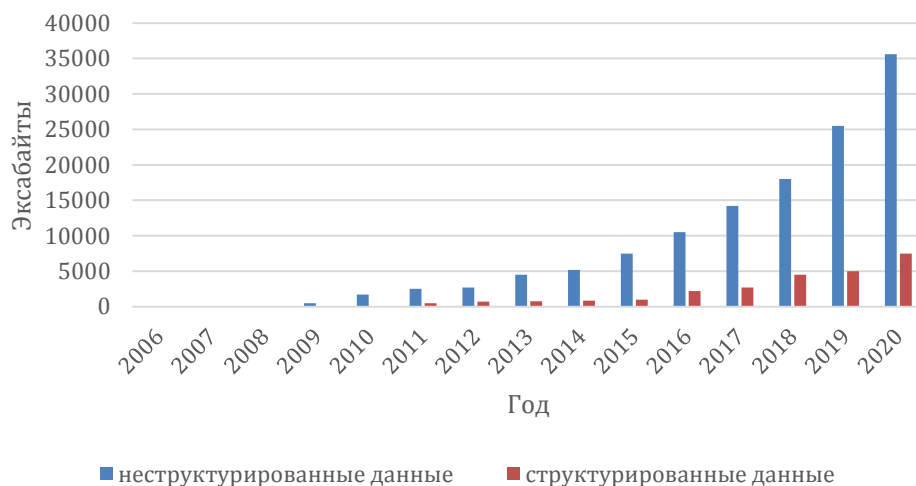


Рис.1 – График роста структурированных и неструктурированных данных

В связи с увеличением нагрузки на ресурсы вычислительных центров всё чаще задействуется такое понятие как облака или облачные вычисления.

**Облачные вычисления** (англ. Cloud computing) — модель обеспечения удобного сетевого доступа по требованию к некоторому общему фонду конфигурируемых вычислительных ресурсов (например, сетям передачи данных, серверам, устройствам хранения данных, приложениям и сервисам — как вместе, так и по отдельности), которые могут быть оперативно предоставлены и освобождены с минимальными эксплуатационными затратами или обращениями к провайдеру.

Актуальность использования облачных вычислений растёт с геометрической прогрессией в связи с рядом некоторых особенностей [2]:

1. **Доступность** – облака доступны всем, из любой точки, где есть Интернет, с любого компьютера, где есть браузер. Это позволяет пользователям (предприятиям) экономить на закупке высокопроизводительных, дорогостоящих компьютеров. Также сотрудники компаний становятся более мобильными, так как могут получить доступ к своему рабочему месту из любой точки земного шара, используя ноутбук, нетбук, планшетный ПК или смартфон.

2. **Низкая стоимость.** Косвенный показатель, однако зачастую не соответствующий действительности. Перечислим факторы, позволившие снизить стоимость использования облаков:

- снижение расходов на обслуживание виртуальной инфраструктуры, вызванное развитием технологий виртуализации, за счет чего требуется меньший штат для обслуживания всей ИТ инфраструктуры предприятия, однако увеличивается квалификация данного штата сотрудников;

- оплата фактического использования ресурсов, пользователь облака платит за фактическое использование вычислительных мощностей облака, что позволяет ему эффективно распределять свои денежные средства. Это позволяет пользователям (предприятиям) экономить на покупке лицензий к ПО;

- развитие аппаратной части вычислительных систем, в связи с чем снижение стоимости оборудования.

3. **Гибкость** – неограниченность вычислительных ресурсов (память, процессор, диски), за счет использования систем виртуализации, процесс масштабирования и администрирования «облаков» становится достаточно легкой задачей, так как «облако» самостоятельно может предоставить необходимые ресурсы.

4. **Надёжность** – надёжность «облаков», особенно находящихся в специально оборудованных центрах обработки данных (ЦОД), очень высокая, так как такие ЦОД имеют резервные источники питания, профессиональных работников, регулярное резервирование данных, высокую пропускную способность Интернет-канала, высокая устойчивость к DDOS атакам.

5. **Безопасность** – «облачные» сервисы имеют достаточно высокую безопасность при должном ее обеспечении, однако при халатном отношении эффект может быть полностью противоположным.

6. **Большие вычислительные мощности** – вы как пользователь «облачной» системы можете использовать все ее вычислительные способности, заплатив только за фактическое время использования. Предприятия могут использовать данную возможность для анализа больших объемов данных.

В целом, **облако** может иметь различное строение. К примеру, может быть просто хранилищем данных, а может иметь в составе один или несколько вычислительных кластеров.

**Вычислительный кластер** – это совокупность компьютеров, объединенных в рамках некоторой сети для решения одной задачи, которая для пользователя представляется в качестве единого ресурса.

Ввиду того, что облачные вычисления используют в различных целях, существует несколько моделей развёртывания облака [3].

– **Частное облако** – инфраструктура, предназначенная для использования одной организацией, включающей несколько потребителей (например, подразделений одной организации), возможно также клиентами и подрядчиками данной организации. Частное облако может находиться в собственности, управлении и эксплуатации как самой организации, так и третьей стороны (или какой-либо их комбинации), и оно может физически существовать как внутри, так и вне юрисдикции владельца.

– **Публичное облако** – инфраструктура, предназначенная для свободного использования широкой публикой. Публичное облако может находиться в собственности, управлении и эксплуатации коммерческих, научных и правительственных организаций (или какой-либо их комбинации). Публичное облако физически существует в юрисдикции владельца — поставщика услуг.

– **Общественное облако** — вид инфраструктуры, предназначенный для использования конкретным сообществом потребителей из организаций, имеющих общие задачи (например, миссии, требований безопасности, политики, и соответствия различным требованиям). Общественное облако может находиться в кооперативной (совместной) собственности, управлении и эксплуатации одной или более из организаций сообщества или третьей стороны (или какой-либо их комбинации), и оно может физически существовать как внутри, так и вне юрисдикции владельца.

– **Гибридное облако** — это комбинация из двух или более различных облачных инфраструктур (частных, публичных или общественных), остающихся уникальными объектами, но связанных между собой стандартизованными или частными технологиями передачи данных и приложений (например, кратковременное использование ресурсов публичных облаков для балансировки нагрузки между облаками).

Так, к примеру, в БГУИР было запущено внутреннее частное гибридное облако для общеуниверситетских компьютерных классов на базе технологии VMware vCenter решением VMware Horizon.

Технология подразумевает работу по различным дисциплинам, предоставляя доступ к полноценной операционной системе или конкретным приложениям, посредством обращения к виртуальному терминальному серверу с использованием тонких клиентов.

Реализации данных технологий подразумевают различные решения для конечных пользователей.

**Виртуализация рабочих столов** (Virtual Desktop Infrastructure) – решение, позволяющее запускать операционную систему внутри виртуальной машины на сервере в ЦОД и работать с ней удаленно с помощью специальных протоколов передачи данных с любого устройства. На сегодняшний момент наиболее популярные решения в VDI – Citrix XenDesktop, VMware View, Microsoft VDI, Quest vWorkspace. Непосредственно в процессе IT-образования в БГУИР используются и применены решения компаний VMware и Huawei FusionCloud.

**Виртуализация приложений** (Application Virtualization) – технология, позволяющая доставлять и выполнять приложения на реальных машинах без привычной установки программ в ОС. Наиболее популярные решения – Microsoft App-V, Citrix XenApp, VMware ThinApp. В БГУИР широко применяется в процессе обучения студентов технология VMware ThinApp, а для работы некоторых структурных подразделений развернуты также приложения на базе программного продукта Citrix XenApp.

Для работы преподавателей и, соответственно, обучения учащихся было установлено два класса по 15 терминальных сессий на виртуальные серверы (termsrvclass9 и termsrvclass11).

Однако при построении столкнулись с рядом проблем.

В связи с тем, что проведение модернизации существующего оборудования лишь незначительно повлияет на качество работы, планируется провести ряд мероприятий, включающий в себя установку дополнительного кластера в состав ЦОД в первом корпусе БГУИР, а также установку дополнительной системы хранения данных (СХД).

Задумка расширения ЦОД в БГУИР зародилась благодаря потребности иметь сконцентрированную вычислительную мощность, но расположенную физически в разных местах.

Как следствие, будет осуществлён переход на тонкие клиенты канцелярии и администрации университета. Для сотрудников это будет удобно, потому как система будет проста в использовании, личные данные сохраняются в пределах своего профиля. Ещё одним, и наиболее значительным, достоинством является возможность работать удалённо. То есть можно получить доступ с чего бы то ни было, имея подключение к сети, к примеру, с планшетного ПК или смартфона. Это существенно упрощает работу, и, в случае отсутствия у сотрудников возможности посетить свои рабочие места, ущерб предприятию либо отсутствует, либо значительно уменьшается (разумеется, всё зависит от выполняемых должностных обязанностей).

Что же касается администрирования, ситуация обстоит следующим образом: обслуживать кластер легче ввиду централизации управления. В связи с этим можно использовать меньшее количество кадров для обслуживания как оборудования, так и виртуальных серверов. Однако здесь встаёт вопрос о высокой квалификации работников.

С другой стороны, цены на серверные комплектующие значительно выше цен на комплектующие для персональных ЭВМ. Такая же ситуация обстоит и с программным обеспечением.

Выгода заключается в следующем: поскольку кластером будет замещаться довольно внушительное количество персональных ЭВМ, суммарное количество лицензий для ПЭВМ будет превосходить по стоимости серверное лицензионное ПО.

Хотя и с лицензированием ПО могут возникнуть неувязки. Существуют программы, в которых процедура лицензирования может проводиться для каждого терминального сеанса в отдельности, что, по сути, является аналогичным лицензированию ПО на персональных ЭВМ.

Если применять данные технологии для организации учебного процесса, нужно будет продумывать всё до мелочей, поскольку в данном вопросе присутствуют некоторые нюансы.

В процессе обучения студентам и преподавателям необходимы различные виды ПО, в зависимости от дисциплины. В связи с тем, что ПО, используемое для организации учебного процесса (проектирования, создания чертежей и т.п.) бывает весьма ресурсоёмким, стоит задуматься об использовании более производительной вычислительной техники, что в свою очередь отрицательно влияет на бюджет.

Говоря о дополнительной СХД, нельзя не заметить ряд преимуществ.

Во-первых, использование дополнительной СХД позволит упростить структуру хранилища, т.е. все данные, равно как и вычислительные мощности, будут сконцентрированы в одном месте. При этом данные между собой будут разграничены так, как необходимо пользователям в соответствии с разграничениями прав доступа.

Во-вторых, СХД будет использоваться не только в качестве хранилища данных сотрудников структурных подразделений университета (и, возможно, студентов), но и для хранения резервных копий, инсталляционного и сервисного ПО, а также учебных материалов для учащихся (библиотечные материалы). Однако и в данном вопросе присутствует проблема, а именно: причина кроется в том, что учащиеся и сотрудники хранят в своих профилях не так уж и мало не значимой, а порой и мусорной информации, и, что самое важное, значительная её часть может совершенно не использоваться. Поэтому придётся расширять хранилище до куда больших размеров вопреки рентабельности.

В итоге, с точки зрения реализации проекта, выход на общепринятые тенденции повысит статус ВУЗа как на республиканской, так и на международной арене, с учетом наличия высококвалифицированных специалистов, которых готовит БГУИР.

**Список использованных источников:**

1. International Data Corporation. Analysts.
2. Каменщиков А.А. Облачные технологии и интероперабельность информационных систем в здравоохранении, Институт радиотехники и электроники им. В.А. Котельникова РАН, Москва, 2013
3. Морозова О.А., Лосева В.В Информационные технологии в государственном и муниципальном управлении. Учебное пособие для бакалавриата и магистратуры 2-е издание, Москва, Юрайт. – 2018. – 131 с.

## ФУНКЦИОНАЛЬНАЯ МОДЕЛЬ ИНФОРМАЦИОННОЙ СИСТЕМЫ АГРЕГАЦИИ БИЗНЕС-ДАННЫХ

Моженкова Е.В., Парамонов А.И.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Парамонов А.И. – канд. техн. наук, доцент

В современном мире, наполненном большим количеством информационных систем, важное место занимает их обработка с целью получения дополнительной информации. Одной из наиболее актуальных проблем является обработка банков данных корпоративных информационных систем для улучшения процесса сопровождения и уменьшения затрат разработчика для локализации дефекта. Поэтому возникла потребность в создании программного средства, которое можно будет использовать для организации процесса обработки, сбора и передачи необходимой части банка данных на тестовую базу разработчика.

Для локализации дефектов работы корпоративных информационных систем (далее КИС) на стороне разработчика предлагается реализовать механизм импорта/экспорта данных клиента. Для этого разрабатывается автоматизированная информационная система агрегации бизнес-данных «Агрегатор данных». С целью обеспечения полноты реализации функциональных требований к системе повышенное внимание было уделено её функциональному проектированию. Параллельное проектирование концептуальной, логической и графической моделей системы выполнялось с помощью языка визуального моделирования UML [1]. Процесс проектирования предоставил возможность последовательно перейти от разработки наиболее общих моделей и представлений концептуального уровня к более частным и детальным представлениям логического и физического уровня. Диаграммы вариантов использования (Use Case) – это важный метод анализа требований, который широко используется в современной разработке программного обеспечения.

Анализ алгоритма обмена данными между КИС клиента и разработчика [2] позволил выделить основных бизнес-акторов системы: разработчик и пользователь, а также основные бизнес-сущности процесса. Диаграмма бизнес-сущностей и их атрибутов представлена на рисунке 1.

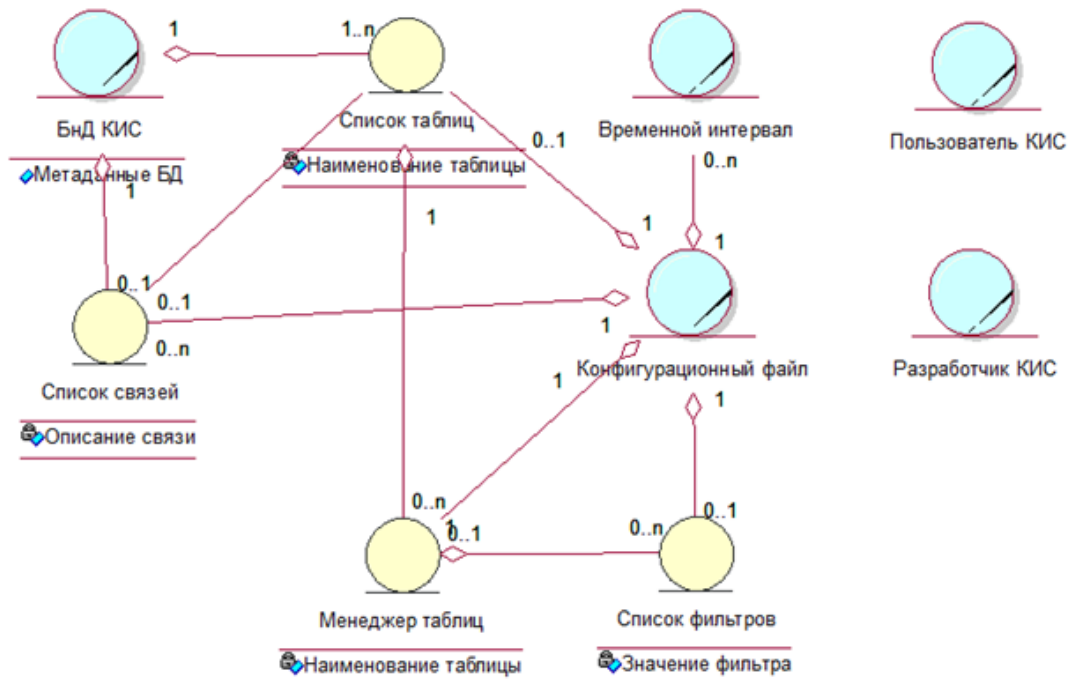


Рисунок 1 – Диаграмма бизнес-сущностей и их атрибутов АИС «Агрегатор данных»

Пользователь КИС ведет непосредственную работу с конечной системой и является потенциальным источником генерации ошибок в КИС в процессе ввода и обработки первичной информации. Разработчик БД КИС является источником сведений о структуре КИС (логика работы системы и хранения данных) и настройки системы. Данной роли отводится дополнительная настройка системы «Агрегатор данных» и конфигурационного файла для передачи данных.

Исходя из потребностей действующих лиц предложены следующие варианты использования: вход в систему, настройка подключения к БД, настройка конфигурационного файла, экспорт данных,



импорт данных. Вариант использования «Войти в систему» не соответствует какой-либо явной цели действующего лица, а описывает исключительно функциональные требования к системе по обеспечению ограничения доступа к данным. Предполагается, что про каждого пользователя система должна знать, какую роль по отношению к ней он выполняет, к каким функциям и данным ему можно предоставлять доступ. Настройка подключения к БД включает процесс установки параметров подключения к БД КИС. Настройка конфигурационного файла включает процесс описания структуры выгружаемых данных в xml-формате [3]. Бизнес-процесс «Экспорт данных» декомпозируется на подпроцессы: «Считывание настройки менеджера таблиц», «Выборка данных из менеджера таблиц», «Определение связей между менеджером и подчиненными таблицами», «Выборка данных из подчиненных таблиц», «Загрузка данных в промежуточные таблицы», «Сохранение выгрузки в файл», «Формирование протокола выгрузки данных». Каждый из этих подпроцессов, а также и взаимодействие, будут описаны на диаграмме деятельности. Бизнес-процесс «Импорт данных» также декомпозируется на подпроцессы: «Выбор конфигурационного файла», «Обработка конфигурационного файла», «Загрузка данных из файла в промежуточные таблицы», «Вставка данных в БД», «Формирование протокола обработки файла». Диаграмма вариантов использования АИС «Агрегатор данных» представлена на рисунке 2.

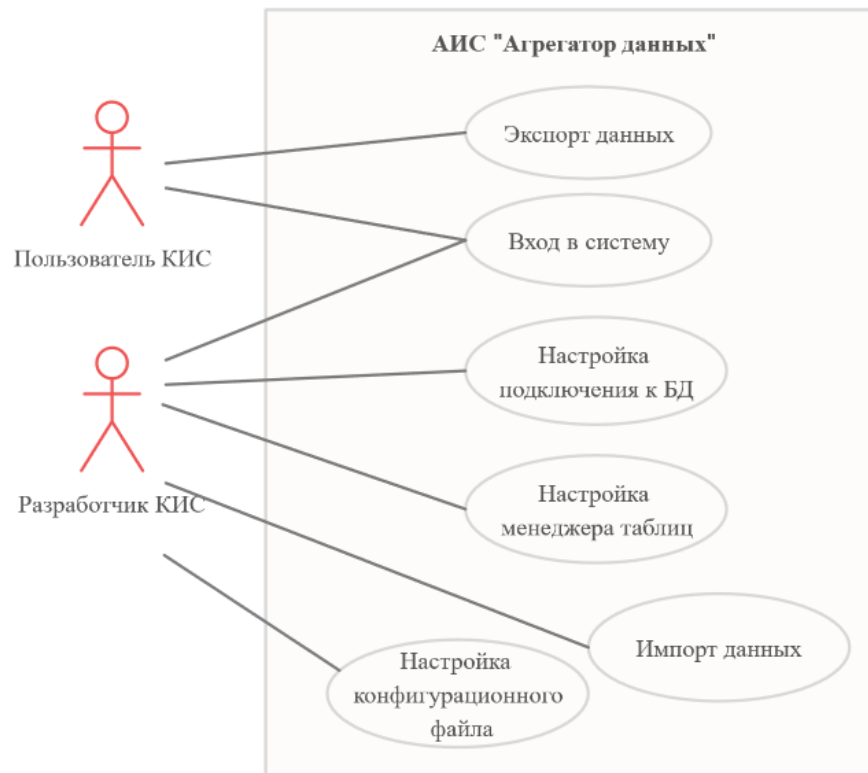


Рисунок 2 – Диаграмма вариантов использования

Построение функциональной модели позволило выявить внутренние и внешние факторы, которые воздействуют на работу системы и которые необходимо учитывать при ее функционировании. Важным шагом моделирования системы является дальнейшее составление сценариев основных вариантов деятельности программного средства. Это позволяет изучить эффективность различных конфигураций применения системы пользователями и отслеживать потоки информационных данных в ней с целью выработки рекомендаций по использованию.

**Список использованных источников:**

1. Ciccozzi, F.: Unicom: a semantics-aware model compiler for optimised predictable software. In: International Conference on Software Engineering (ICSE) 2018-New Ideas and Emerging Results (NIER), May 2018. UML, Alf, fUML, compilation, model-driven engineering, predictability, semantics.
2. Моженкова, Е. В. Метод агрегации потоков данных в задачах локализации проблем / Моженкова Е. В., Парамонов А. И. // Информационные технологии и системы 2019 (ИТС 2019) = Information Technologies and Systems 2019 (ITS 2019) : материалы международной научной конференции, Минск, 30 октября 2019 г. / Белорусский государственный университет информатики и радиоэлектроники; редкол. : Л. Ю. Шилин [и др.]. – Минск, 2019. – С. 242 – 243.
3. Моженкова, Е. В. Схема генерации и восстановления данных при локализации дефекта / Е. В. Моженкова, А. И. Парамонов // Компьютерные системы и сети: 55-я юбилейная научная конференция аспирантов, магистрантов и студентов, Минск, 22-26 апреля 2019 г. / Белорусский государственный университет информатики и радиоэлектроники. – Минск, 2019. – С. 132 – 133.

## РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ СИСТЕМЫ 1С-BITRIX

Мячков А. А., Деменковец Д. В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Деменковец Д. В. – Старший преподаватель

В работе рассматривается система управления контентом (CMS) 1С-Bitrix. Описываются основные возможности данной платформы, а также преимущества и недостатки данной системы при создании веб-приложений.

Система управления контентом (англ. CMS - Content Management system) 1С-Bitrix представляет собой среду для разработки веб-приложений, то есть каркас для создания программных продуктов. На данный момент одно из лучших коммерческих решений в Республике Беларусь и Российской Федерации. В данном докладе рассматривается платформа 1С-Bitrix, описываются положительные стороны и причины ее использования.

В 2019 году компанией iTrack было проведено исследование, которое показало, что более 12% из 5 миллионов сайтов доменной зоны .ru использует платформу 1С-Bitrix, а именно около 550 тысяч веб-приложений, как отображено на рисунке 1 [1].

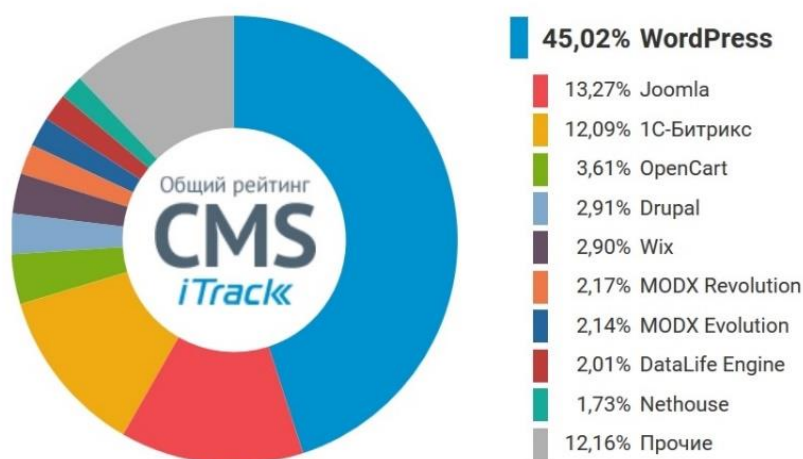


Рисунок 1 – Общий рейтинг систем управления контентом (CMS)

Вторая диаграмма на рисунке 2 демонстрирует рейтинг самых популярных платных систем управления контентом, из которого видно, что более чем на 53 процента сайтов установлена среда разработки 1С-Bitrix [2].

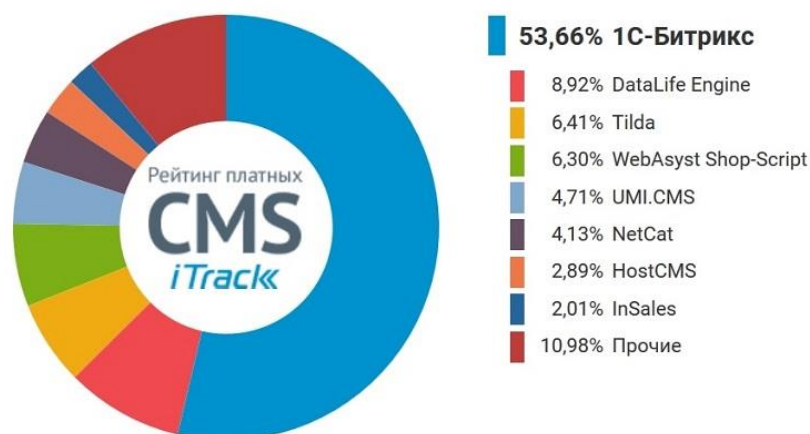


Рисунок 2 – Рейтинг платных систем управления контентом (CMS)

В результате исследования были выявлены причины, почему система 1С-Bitrix лидирует среди ему подобных систем.

Первой важной причиной является широкая функциональность, которую легко использовать. В

систему входит множество инструментов и модулей.

Среди основных возможностей данной среды разработки выделяют [3]:

1) Информационные блоки, через которые легко обновлять контент веб-приложений, а также осуществлять импорт и экспорт данных.

2) Документация, с помощью которой можно настроить цепь публикации новостей, а также видеть историю изменений. Это лучше всего подходит, для организации работы различных онлайн-магазинов и виртуальных редакций.

3) Поддержка множества веб-серверов, разнесенных географически для достижения максимальной скорости отдачи контента клиенту, который ускоряет скорость загрузки веб-страниц (англ. CDN - Content Delivery Network).

4) Настраиваемая автоматическая рассылка, наиболее удобна для интернет-магазинов.

5) Доступна технология, позволяющая быстро разрабатывать мобильные приложения под мобильные платформы iOS и Android [4].

В платформе 1С-Bitrix заложены все основные решения для большего количества сайтов, которые просто настраивать под конкретное решение. Это делает разработку веб-приложений очень быстрой [3].

Вторая причина преимущества системы 1С-Bitrix — это работа с целевой аудиторией в виде представителей бизнес-среды. Это приводит к стабильности, надежности и простоте на коммерческом рынке.

Третья причина – безопасность. Заключается в защите для многих компаний защита от угроз и вмешательства посторонних лиц. Является ключевым фактором при выборе системы управления контентом. Например, CMS WordPress – является одним из «лидеров» по количеству взломов.

Основные механизмы защиты данных:

1) Веб-антивирус, который проверяет файлы и выявляет файлы и участки кода, которые могут быть опасными. Таким образом администратор имеет возможность их быстро устранить.

2) Защищенная авторизация. Шифрование работает даже без активного SSL сертификата безопасности.

3) Облачные хранилища, позволяющие восстановить потерянные файлы, а также позволяют автоматически выгружать необходимые файлы.

4) Журнал активности, регистрирующий все попытки взлома.

Отдельно необходимо выделить то, что разработка ведется с использованием языка программирования PHP. PHP — скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков, применяющихся для создания динамических веб-сайтов [5]. Этот язык очень прост в освоении.

Однако среди всех преимуществ есть и недостатки, основные из которых:

1) Высокая стоимость для нестандартных решений, так как эта платформа создана под типичную функциональность веб-приложений.

2) Требовательность к ресурсам сервера, что особенно важно при большой посещаемости веб-приложения.

3) Необходимость ежегодной оплаты обновлений [3].

Подводя итог, можно утверждать, что система управления контентом 1С-Bitrix хорошее коммерческое решение, если необходимо быстро создать веб-приложение, которое будет защищенным и легко управляемым с поддержкой журналов всех действий и изменений, внесенных сотрудниками. Тем не менее, это платный продукт, который потребует еще больших инвестиционных вложений, если необходимо разработать нестандартное веб-приложение.

#### **Список использованных источников:**

1. Рейтинг CMS за 2019 год // Информация о компании iTrack [Электронный ресурс]. – 2020. – Режим доступа: <https://itrack.ru/research/cmsrate/> Дата доступа: 20.03.2020

2. CMS Рейтинг // Обзор возможностей CMS 1С-Bitrix [Электронный ресурс]. – 2020. – Режим доступа: <https://cms-rating.ru/cms-shop-1c-bitrix/> Дата доступа: 23.03.2020

3. Что такое Bitrix // Habr [Электронный ресурс]. – 2016. – Режим доступа: <https://habr.com/en/company/trinion/blog/315538/> Дата доступа: 23.03.2020

4. 1С-Битрикс: Управление сайтом // 1С-Битрикс Республика Беларусь [Электронный ресурс]. – 2020. – Режим доступа: <https://www.1c-bitrix.by/products/cms/> Дата доступа: 23.03.2020

5. PHP is a popular general-purpose scripting language that is especially suited to web development. // [www.php.net/](http://www.php.net/) [Электронный ресурс]. – 2020. – Режим доступа: <https://www.php.net/> Дата доступа: 23.03.2020

# РАСШИРЕНИЕ МОДЕЛИ ОБНАРУЖЕНИЯ УЯЗВИМОСТЕЙ В WEB-ПРИЛОЖЕНИЯХ, ОСНОВАННОЙ НА СТАТИЧЕСКОМ АНАЛИЗЕ ИСХОДНЫХ КОДОВ

Оношко Д.Е.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Бахтизин В.В. – канд. техн. наук

Предлагается способ расширения системы оценки для ранее описанной модели обнаружения уязвимостей в web-приложениях, основанной на статическом анализе исходных кодов.

По данным Открытого проекта безопасности web-приложений, в настоящее время наиболее значимой угрозой безопасности web-приложений являются инъекции [1]. При этом наиболее ценной для злоумышленника целью атаки являются запросы к системам управления базами данных (СУБД), использующимся подавляющим большинством современных web-приложений.

Существующие технические решения, такие как объектно-реляционное отображение (ORM), хранимые процедуры (stored procedures) или подготовленные выражения (prepared statements) позволяют существенно усложнить проведение подобной атаки злоумышленником, однако, как и экранирование полученных извне данных, не предотвращают возникновение уязвимостей в полной мере, поскольку обеспечивают требуемый эффект исключительно при грамотном применении, т.е. отсутствии ошибок со стороны разработчика. Поскольку подобные решения накладывают ряд ограничений, связанных с границами их применимости, также оказывается невозможным исключить обращения к СУБД в обход подобных решений в нестандартных задачах, а значит, и сопутствующие ошибки их применения.

Таким образом, важной мерой обеспечения качества web-приложений оказывается контроль их исходных кодов на предмет наличия ошибок, обуславливающих уязвимость к SQL-инъекциям.

Основой предложенного в [2] метода оценки качества web-приложений является модель обнаружения уязвимостей, предполагающая разделение анализируемого исходного кода на множество обобщённых процедур  $P = \{p_1, p_2, \dots, p_n\}$ , где  $p_i$  характеризуется множеством оценок  $m_i = M(p_i)$  её формальных параметров с точки зрения производимых процедур преобразований данных, с последующим анализом передачи данных между этими процедурами. Предлагаемая бинарная система оценки позволяет обеспечить назначение оценок всем элементам путём упрощённой абстрактной интерпретации [3] исходного кода при сохранении невысокой сложности программной реализации метода обнаружения. Основной проблемой практического применения модели с бинарной системой оценки является большое количество ложных срабатываний (ввиду принципов назначения оценок — преимущественно ложноположительных).

Одним из возможных подходов для снижения доли ложноположительных результатов является расширение системы оценки таким образом, чтобы значения оценок соответствовали не только предполагаемому подмножеству значений соответствующего элемента данных (что соответствует оценкам S и U в модели [2]), но и другим свойствам элементов данных. Помимо ручного выбора элемента данных как имеющего оценку S (оценка UDS в исходной модели), примером таких свойств может быть, например, происхождение элемента данных и его оценки:

- данные, полученные извне;
- данные, полученные как значение литерала в исходном коде;
- данные, полученные из результатов выполнения запроса к СУБД;
- данные, полученные в процессе обработки внутри web-приложения;
- и т.п.

В некоторых случаях учёт этих свойств можно осуществить путём расширения базовой бинарной шкалы оценок. В общем же случае целесообразно рассматривать оценку  $m_i$  элемента данных как вектор  $\vec{m}_i = (m_{i1}, m_{i2}, \dots)$ , каждый элемент которого соответствует одному из свойств.

В качестве примера такого расширения системы оценки можно рассмотреть следующий набор свойств:

- **возможность прямой подстановки данных в запрос:** U — подстановка приводит к уязвимости, S — подстановка допускается;
- **источник элемента данных:** Request — данные, полученные от пользователя вместе с запросом, Literal — данные, представленные литералом в исходном коде web-приложения, Database — данные, полученные из базы данных web-приложения, Derived — данные, полученные их

других данных в ходе обработки.

Нетрудно заметить, что некоторые комбинации значений свойств могут не иметь смысла. Так, например, строковые литералы, записанные в исходном коде, как правило, могут передаваться в процедуры, выполняющие запрос к СУБД, без ограничений, поскольку при этом не происходит изменения логики запроса. Тем не менее, комбинация (U, Literal) возможна, если при программной реализации метода обнаружения следует учесть возможность использования строковых литералов из недоверенных вспомогательных модулей web-приложения. Другая подобная комбинация свойств — (S, Request).

Следует понимать, что ключевым свойством в такой расширенной системе оценки по-прежнему остаётся свойство, отражающее возможность или невозможность прямой подстановки данных в запрос: именно по сочетанию значений этого свойства для формального параметра обобщённой процедуры и для передаваемого фактически элемента данных можно судить о наличии или отсутствии уязвимости. Однако сведения об источнике данных могут использоваться для назначения более точных оценок результатам некоторых операций и стандартных процедур. Так, например, результат конкатенации двух строковых значений в общем случае имеет оценку вида (U, \*), где \* — произвольное значение свойства, однако при конкатенации, например, двух элементов с оценками (S, Literal) результат получит оценку (S, Derived).

Дальнейшее расширение модели за счёт перехода к оценкам-векторам может заключаться в увеличении количества учитываемых свойств, а также в поддержке одновременного обнаружения уязвимостей к различным видам инъекций, а не только к SQL-инъекциям. Однако на практике увеличение размерности оценочного вектора приводит к повышению сложности анализатора, в том числе в связи с тем, что некоторые добавляемые свойства могут потребовать реализации дополнительных настроек анализа.

Одним из возможных способов сохранения баланса между сложностью анализа и достоверностью получаемых результатов при использовании предлагаемого способа расширения системы оценки является отложенное вычисление значений дополнительных свойств по упрощённой схеме.

Так, например, при анализе web-приложения, использующего несколько баз данных, может оказаться полезной информация не только о том, что путь к эксплуатации уязвимости связан с использованием данных, полученных от СУБД, но и о том, какая именно база данных выступает в качестве источника этих данных. При этом пользователю анализатора может быть предоставлена возможность отметить одну из баз данных как доверенную. Добавление информации о том, какая именно база данных является источником элемента данных, в оценочный вектор будет в этом случае нецелесообразным, поскольку сам по себе идентификатор или условный номер базы данных не несёт полезной информации для анализа.

Для приведённой выше системы оценки в подобной ситуации более простым решением будет добавление специального значения Trusted database для оценки данных, поступающих из доверенной базы данных. При этом информация о том, какая именно база данных является источником, которая потребует только при формировании отчёта для пользователя анализатора, может быть извлечена из структур данных, используемых для представления структуры анализируемого web-приложения при восстановлении пути инъекции.

Таким образом, предлагаемый способ расширения модели обнаружения уязвимостей при сохранении сравнительно невысокой сложности программной реализации обнаружения уязвимостей позволяет уменьшить долю ложноположительных результатов и упростить отслеживание путей эксплуатации найденных уязвимостей.

**Список использованных источников:**

1. OWASP Top 10 – 2017 [Электронный ресурс]. — Режим доступа: [https://wiki.owasp.org/images/7/72/OWASP\\_Top\\_10-2017\\_%28en%29.pdf.pdf](https://wiki.owasp.org/images/7/72/OWASP_Top_10-2017_%28en%29.pdf.pdf). — Дата доступа: 20.03.2020.
2. Оношко Д.Е., Бахтизин В.В. Метод оценки качества web-приложений, основанный на обнаружении уязвимостей // Цифровая трансформация. 2018. №1(2). С. 58–65.
3. Cousot P., Cousot R. Abstract interpretation: a unified lattice model for static analysis of programs by construction or approximation of fixpoints // Conference record of the Fourth ACM symposium on principles of programming languages. — 1977. — p. 238–252.

## ОБЗОР ВОЗМОЖНОСТЕЙ ПЛАТФОРМЫ FLUTTER

*Петровская В.В., Деменковец Д.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Деменковец Д.В. – старший преподаватель*

В тексте работы анализируются возможности платформы Flutter SDK. Описываются преимущества данной платформы. Выполняется сравнение подходов к разработке и применение для создания приложений с использованием технологии Flutter.

Flutter SDK (от англ. software development kit) – бесплатное кроссплатформенное средство от Google с открытым исходным кодом для быстрой разработки мобильных приложений для iOS и Android. Позволяет быстро и качественно создавать пользовательские интерфейсы [1]. Был представлен Flutter в 2017 году, официально выпущен в декабре 2018 года компанией Google. Данная технология позволяет разработчикам программного обеспечения создавать кроссплатформенные приложения, которые можно запускать на различных операционных системах, таких как iOS или Android. Логотип платформы Flutter приведен на рисунке 1.

Главной целью разработчиков Flutter SDK было создать платформу проектирования пользовательских интерфейсов для любого размера экрана за счет поддержки технологии для мобильных (mobile applications), настольных (desktop applications), веб-приложений (web applications) и встраиваемых систем (embedded systems). Платформа получила широкое распространение еще до выхода первой версии. К примеру, наиболее популярные приложения, использующие технологию Flutter – Xianyu, GoogleAds, Hamilton Musical app, Reflectly [3].



Рисунок 1 – Логотип платформы Flutter

В 2015 году компания Facebook представила миру платформу для разработки мобильных приложений React Native. Ответом от Google явилось создание платформы Flutter, была представлена в 2017-ом. Обе технологии являются кроссплатформенными, однако подходы к разработке отличаются в значительной степени. React Native использует мост JavaScript. Данный мост служит для связи с модулями платформы и является так называемой прослойкой между приложением и устройством. Данный подход в значительной степени снижает производительность приложения. В платформе Flutter мост отсутствует, Flutter выполняет отрисовку визуальных компонентов пользовательского интерфейсов «сам», а обращение к сервисам устройства происходит через так называемый Platform Channels, буфер обмена между платформой и кодом программы [3].

Платформа Flutter использует язык программирования Dart. Команда разработчиков этого языка пыталась использовать больше десятка языков программирования, включая JavaScript, Objective-C и Java, однако остановилась на языке Dart, который был создан Google в октябре 2011. Dart позиционируется в качестве альтернативы JavaScript. Язык долгое время не пользовался популярностью среди разработчиков, и в 2018 году и вовсе был назван самым худшим языком для изучения. Однако Dart имеет ряд преимуществ, которые и обусловили такой выбор разработчиков Flutter.

Язык программирования Dart позволяет работать с JIT компиляцией и делать горячее развертывание или установку приложения (hot reload) во время разработки. Благодаря возможности обновления приложения с сохранением состояния в значительной степени сокращается цикл разработки приложения. Во время выполнения сборки итогового приложения (release build) код компилируется непосредственно в машинные инструкции, которые выполняются на устройстве. За счет этого достигается скорость выполнения операций, сравнимая с Objective-C, Swift, Java или Kotlin. Язык Dart обладает опциональной, строгой типизацией, поддерживает реактивный стиль программирования, обладает хорошей поддержкой управления объектами с помощью прогрессивного сборщика мусора. Кроме этого Dart обладает схожим с языком Java синтаксисом и прост в освоении.

Платформа Flutter стремится обеспечить производительность 60 кадров в секунду (fps) или производительность 120 кадров в секунду на устройствах с частотой обновления экрана 120 Гц. Отображением занимается графическая библиотека под названием Skia. Это быстрая 2D графическая библиотека, разработанная на языке C++. Кроме графической Skia под капотом у Flutter текстовый движок, заимствованный из Android. Это позволяет быстро обновлять исходный код при

внесении изменений [3].

Flutter не использует визуальные элементы интерфейса платформы, он предоставляет свои собственные. Создатели Flutter SDK разработали свои пакеты этих интерфейсов на языке Dart. Flutter отвечает за прорисовку каждого пикселя, за запуск анимации, за обработку жестов. Благодаря этому достигается максимальное сходство в работе на различных устройствах и минимальное влияние операционной системы на итоговый внешний вид приложения. Для дополнительной производительности в работе пользовательского интерфейса визуальные элементы перерисовываются в случае необходимости. Например, когда изменился внешний вид.

Для построения пользовательского интерфейса (user interface) во Flutter используется декларативный подход. В распоряжении разработчиков два набора визуальных элементов для основных мобильных платформ: Material для Android и Cupertino для iOS. Таким образом, разработчики Flutter заново разработали и выполнили отображение на устройствах все пользовательские визуальные элементы (UI-компоненты), полностью повторив их поведение. Непосредственное аппаратное взаимодействие устройства (географическая локация, звук, bluetooth, GPS, акселерометр) с платформой происходит через Platform Channels [4].

Приложения разработанные с помощью платформы Flutter легко тестировать. Технология поддерживает модульное тестирование (unit testing), тестирование визуальных компонентов пользовательского интерфейса (widget testing) и интеграционное тестирование. Модульные тесты выполняют тестирование одиночных функций, методов или классов. Тестирование визуальных компонентов позволяет убедиться, что пользовательский интерфейс выглядит и работает корректно. Интеграционные тесты проверяют взаимодействие модулей и всего приложения в целом.

Интерфейс командной строки платформы Flutter позволяет настраивать концепцию непрерывной интеграции и доставки CI/CD (англ. Continuous integration & Continuous delivery). Концепция, применяемая для выявления нарушений в работе приложения при внесении изменений. Для приложений Flutter имеется встроенную поддержку CI/CD, где используется Nevercode - платформа на основе облачных вычислений.

Flutter позволяет создавать сложные анимации, добавлять в приложения локализацию и менять ее в режиме выполнения приложения (run time). Технология активно развивается, разработчики создают еще больше готовых библиотек, которые легко подключаются в разрабатываемый проект. Не маловажным плюсом Flutter является документация, подробная информация служит отличным помощником разработчикам приложений.

В настоящее время компания Google разрабатывает новую операционную систему Fuchsia. Fuchsia использует платформу Flutter в качестве движка прорисовки пользовательского интерфейса. Google также планирует адаптировать все существующие приложения для Android на Fuchsia (включая среду выполнения приложений платформы Android). Особенностью разрабатываемой системы является способность работать на любой архитектуре центрального процессора. В основе этой системы – микроядро Zircon от Google, которое способно работать на смартфонах, настольных компьютерах и встроенных системах.

Flutter – это новая, современная и актуальная технология. Приложения со временем устаревают, появляются новые идеи и решения. Flutter позволяет воплотить идею в готовый продукт с быстрой скоростью и меньшими ресурсами. Платформа находит отклик и вызывает симпатию как у разработчиков, так и у дизайнеров и бизнеса. Flutter позволяет контролировать каждый пиксель на экране и создавать одинаковый интерфейс для разных устройств, для разработки приложения, как правило, необходима только одна команда разработчиков и относительно небольшие временные и материальные ресурсы.

Существует много источников, позволяющие найти нужную и понятную информацию о Flutter: хорошая документация, простые и информативные руководства, лаконичные и наглядные видео из серии «Flutter Widget of the Week», увлекательное и необычное шоу «The Boring Flutter Development Show» и многое другое.

Flutter – молодая технология, поэтому имеет небольшое, но активно развивающееся сообщество. Можно сказать, что данная технология «ловит волну успеха», вдохнув вторую жизнь в язык программирования Dart. Разработчики платформы предлагают новые интересные архитектурные решения и эти факторы предвещают длительное и успешное существование этой платформы.

**Список использованных источников:**

1. Flutter уроки. От идеи до приложения // Flutter уроки [Электронный ресурс]. – 2020. – Режим доступа: <https://flutter.su/> Дата доступа: 20.03.2020
2. Flutter is Google's UI toolkit for building beautiful, natively compiled applications for mobile, web, and desktop from a single codebase. // Flutter [Электронный ресурс]: веб-сайт – Режим доступа: <https://flutter.dev> Дата доступа: 23.03.2020
3. Дыцяк, В. Стоит ли инвестировать во Flutter. Сравнение Flutter и React Native [Электронный ресурс] / В. Дыцяк // Портал сообщества разработчиков. – Режим доступа: <https://dou.ua/lenta/articles/flutter-for-mobile-apps>. – Дата доступа: 23.03.2020
4. Jagtap, S. Flutter vs React Native: A Developer's Perspective [Электронный ресурс] / S. Jagtap // Nevercode blog. – Режим доступа: <https://nevercode.io/blog/flutter-vs-react-native-a-developers-perspective>. – Дата доступа: 23.03.2020

# ПРОГРАММНОЕ СРЕДСТВО СОХРАНЕНИЯ И ОБРАБОТКИ ТЕХНИЧЕСКИХ ПАРАМЕТРОВ ФРОНТАЛЬНОГО ПОГРУЗЧИКА НА ОСНОВЕ ТЕХНОЛОГИИ MICROSOFT AZURE

Деменковец Д.В., Поваляева Н.И.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Деменковец Д. В., старший преподаватель

В докладе рассматривается программное средство обработки и сохранения технических параметров, полученных с устройства установленного на стреле фронтального погрузчика. Также описывается работа программного средства по передаче и сохранению полученных данных на основе облачной технологии Microsoft Azure с последующей предобработкой и обработкой с использованием алгоритмов цифровой обработки сигналов.

В настоящее время является актуальным контроль и учет перевозимых грузов специальной техникой. Программное средство позволит эффективно выполнять контроль за параметрами этой техники с помощью облачного хранилища, что позволит получить доступ к параметрам удаленно.

В основе разрабатываемого программного обеспечения лежит идея программного средства сбора технических параметров, их передача и сохранение в облачном хранилище Microsoft Azure с последующей предобработкой и обработкой. Для данного программного средства техническими параметрами являются данные полученные с устройства установленного на стреле погрузчика [1]. Это «сырые» данные получаемые с двух акселерометров, двух гироскопов и двух датчиков давления в гидроцилиндрах погрузчика.

Работа ПС заключается в следующем. Значения с параметров передаются и загружаются в облачное хранилище. Программное средство выполняет выгрузку этих данных после этого выполняется предобработка (применение цифровых фильтров), обработка (выполнение дискретного преобразования Фурье) данных для более удобного представления полученных значений в виде спектра, с последующей визуализацией в виде графиков, для более детального анализа работы фронтального погрузчика в целом.

Данное программное средство разработано на основе технологий Microsoft .NET Core и Microsoft Azure, ориентировано для удаленного использования [2].

Для отладки и тестирования программного средства использовалась прототип подъемной стрелы, распечатанной на 3d принтере.

Модель была создана на основе схемы фронтального погрузчика Doosan DL200 [3] и в значительной степени повторяет работу реального механизма. Схема фронтального погрузчика и его модель, созданная для печати на 3d принтере изображены на рисунке 1.

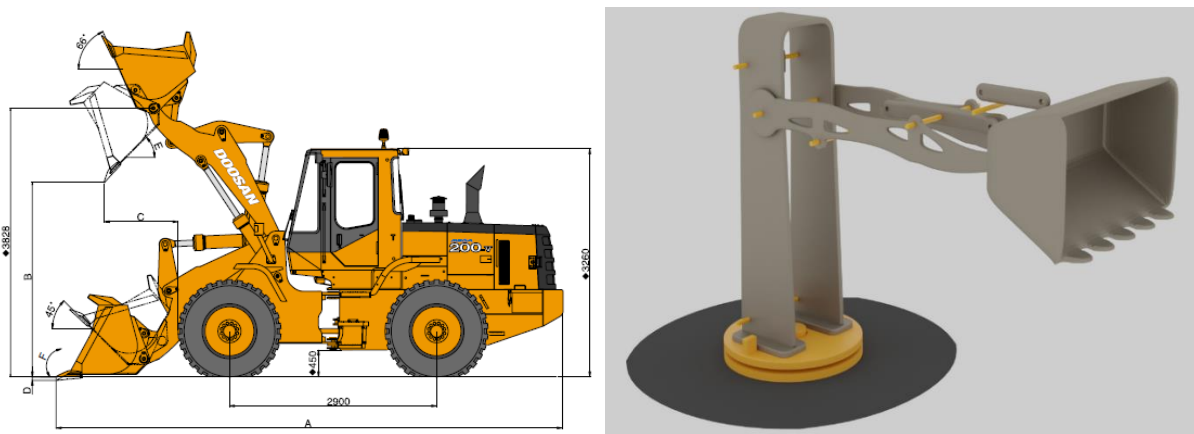


Рисунок 1 – Схема фронтального погрузчика Doosan DL200 [2] и 3d модель его подъемной стрелы

#### Список использованных источников:

1. Кошелев, Д. М. Программное средство сбора и анализа технологических параметров стрелы фронтального погрузчика / Д. М. Кошелев // Компьютерные системы и сети: 55-я юбилейная научная конференция аспирантов, магистрантов и студентов, Минск, 22-26 апреля 2019 г. / Белорусский государственный университет информатики и радиоэлектроники. – Минск, 2019. – С. 112 – 113.
2. Базы данных. Полностью управляемые интеллектуальные службы баз данных // Microsoft Azure [Электронный ресурс]. – 2020. – Режим доступа: <https://azure.microsoft.com/ru-ru/product-categories/databases/> Дата доступа: 20.03.2020
3. Фронтальный погрузчик Doosan Mega 200 // ПСМ Техно [Электронный ресурс]. – 2020. – Режим доступа: [http://псмт.рф/spectehnika/pogruzchiki/frontalnye\\_pogruzchiki/frontalnyj\\_pogruzchik\\_doosan\\_mega\\_200/](http://псмт.рф/spectehnika/pogruzchiki/frontalnye_pogruzchiki/frontalnyj_pogruzchik_doosan_mega_200/) Дата доступа: 20.03.2020



# ПРОГРАММНОЕ СРЕДСТВО СБОРА И АНАЛИЗА ТЕХНОЛОГИЧЕСКИХ ПАРАМЕТРОВ СТРЕЛЫ ФРОНТАЛЬНОГО ПОГРУЗЧИКА

Кошелев Д.М.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Деменковец Д.В. – ассистент

Эксплуатация спецтехники связана с постоянными высокими нагрузками. Подобные условия приводят к значительному увеличению расхода горючего. Чрезмерная эксплуатация транспортного средства и его нецелевое использование приводит к быстрому износу дорогостоящей техники. Программное средство сбора и анализа параметров стрелы фронтального погрузчика помогает решать подобного рода проблемы.

Программное средство сбора и анализа технологических параметров стрелы фронтального погрузчика реализует основные функции, необходимые для сбора и анализа данных о работе техники, к примеру это расчет наработки техники и количества перемещенного груза за смену, неделю, месяц. Это значение очень важно при расчете себестоимости одного часа работы машины [1].

Данное программное обеспечение разработано под платформу Android. Это позволяет его использовать на мобильных устройствах, телефонах и планшетах без необходимости использовать персональный компьютер.

Совместно с программным средством используется специальное аппаратное устройство с датчиками угла наклона и давления в гидравлических цилиндрах стрелы фронтального погрузчика. Один датчик наклона крепится на стрелу, а второй на корпус фронтального погрузчика. Два датчика давлений крепятся в гидравлическую систему подъема. Именно с помощью этих датчиков будет рассчитываться вес перевозимого груза [2].

Разработанное программное средство сбора и анализа параметров работы фронтального погрузчика выполняет следующие функции:

- получение значений с датчиков угла наклона и давлений в гидроцилиндрах;
- вывод значений датчиков в режиме реального времени;
- вывод значений датчиков наклона в виде графиков в режиме реального времени;
- архивирование значений с датчиков в базу данных;
- отображение почасовой суммарной нагрузки на технику за текущие сутки.

Графический интерфейс приложения представлен на рисунке 1:

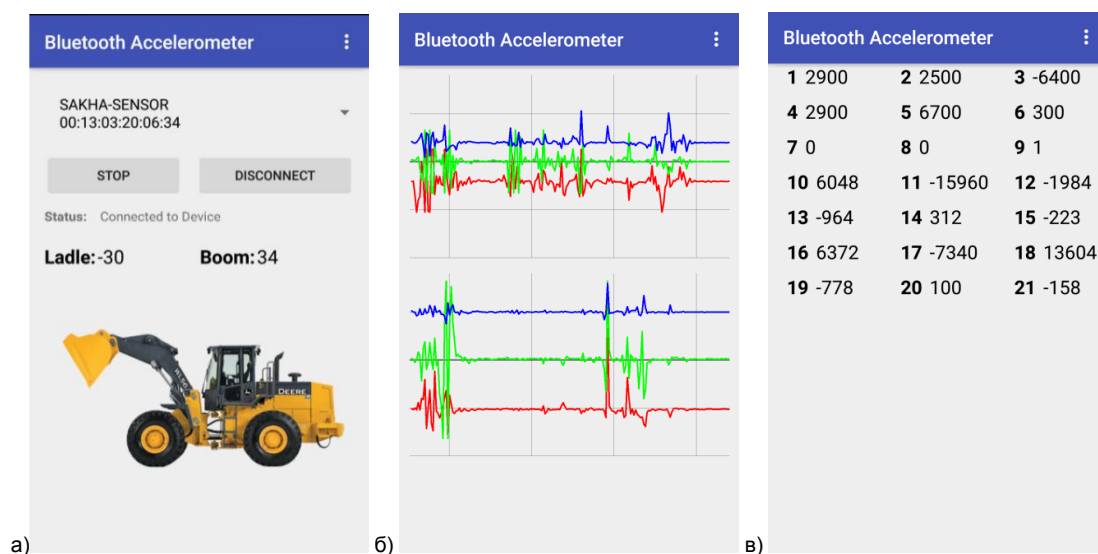


Рисунок 1 – Графический интерфейс ПС: а) главный экран ПС; б) экран графиков реального времени; в) экран текущих значений с датчиков

В дальнейшем планируется дополнительно добавить функции сохранения статистики с помощью облачной технологии (напр. Google Drive). Проблему вибрации планируется решить использованием более совершенных фильтров и выполнением предобработки данных с датчиков.

**Список использованных источников:**

1. Максименко А.Н. ТЕХНИЧЕСКАЯ ЭКСПЛУАТАЦИЯ СТРОИТЕЛЬНЫХ И ДОРОЖНЫХ МАШИН / А. Н. Максименко, В. В. Кутузов; под ред. И.В. Тургель. – Минск : Победа, 2015. – 255 с.
2. АНАЛИЗ НАГРУЖЕННОСТИ ГИДРОЦИЛИНДРА МЕХАНИЗМА ПОДЪЕМА ТЕЛЕСКОПИЧЕСКОЙ СТРЕЛЫ ПОГРУЗЧИКА / М.И. Жилевич, П.Н. Кишкевич, А.А. Зубрицкий // Вестник Костромского государственного университета. Серия тех. науки – 2017. – №12 – С. 31-34.

## ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ ПОСТРОЕНИЯ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ

*Потараев В.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Серебряная Л.В. – канд. техн. наук*

Автоматизированная обработка текстовой информации применяется при решении различных задач, к которым относятся выбор текста для изучения, поиск информации в тексте и др. Один из инструментов для решения этих задач – нейронные сети. В данной работе рассмотрены недостатки обучения искусственной нейронной сети методом градиентного спуска. Предложен генетический алгоритм построения и обучения нейронной сети. Он может быть использован для получения нейронной сети, которая выдаёт результаты с меньшей среднеквадратичной ошибкой, чем сеть, обученная методом градиентного спуска.

За последние десятилетия человечеством создано большое количество текстов, в том числе книг, журналов, сайтов и др. При работе с информацией довольно часто возникает необходимость найти некоторую информацию во множестве документов. Например, студент может искать подходящую книгу по изучаемой теме, искать определение некоторого понятия в учебниках и др. Многие книги и учебники представлены в цифровом виде.

Для автоматизированной обработки текстовой информации, её классификации, а также для решения приведённых задач можно использовать алгоритмы, которые основаны на применении искусственных нейронных сетей. При построении нейронной сети важным является подбор таких её параметров, которые являются оптимальными для решения поставленной задачи. Как правило, множество вариантов структуры и параметров нейронной сети очень велико.

При поиске оптимальных решений в условиях невозможности полного перебора вариантов довольно часто применяют генетический алгоритм [1].

Целью данной работы является разработка методов оптимального построения нейронной сети, основанных на генетическом алгоритме.

Искусственная нейронная сеть (ИНС) - это математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма [2].

Применение нейронной сети состоит из следующих основных этапов:

1. Выбор структуры сети.
2. Обучение сети (подбор весов связей).
3. Использование сети.

На этапе обучения сети обычно происходит подбор весов связей, которые позволяют минимизировать значение некоторой функции (например, среднеквадратичного отклонения выходных сигналов от эталонных значений). Для выбора значений весов, при которых функция минимальна, может быть использован метод градиентного спуска. В случае нейронной сети «многослойный персептрон» (рисунок 1) для минимизации ошибки обычно используется метод обратного распространения ошибки (англ. backpropagation) [3].

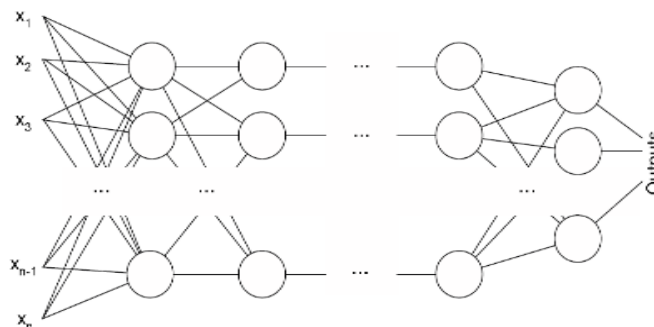


Рисунок 1 – Многослойный персептрон

Метод обратного распространения ошибки подразумевает осуществление градиентного спуска вниз по поверхности функции ошибки, непрерывно подстраивая веса в направлении к минимуму.

Поверхность функции ошибки сложной сети может содержать множество впадин. В связи с этим одним из основных недостатков градиентного спуска является то, что в процессе обучения значения весов связей в сети могут попасть в локальный минимум (неглубокую долину), когда имеется гораздо более глубокий минимум (рисунок 2) [3].

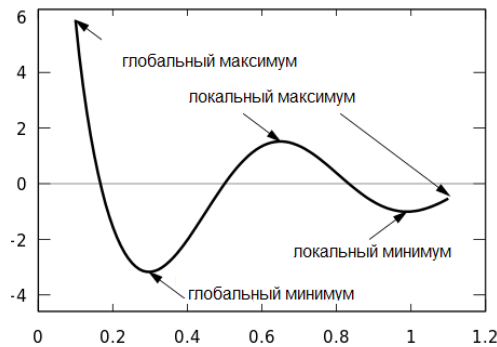


Рисунок 2 – Два минимума функции ошибки нейронной сети

Рассмотрим альтернативу применения градиентного спуска при обучении сети – генетический алгоритм. Генетический алгоритм в общем случае предполагает выполнение следующих шагов [4]:

1. Случайным образом сгенерировать набор пробных решений.

2. Оценить приспособленность текущего поколения. Если приспособленность достаточна, то прекратить выполнение алгоритма.

3. Сгенерировать новое поколение посредством селекции, скрещивания и мутаций.

Рассмотрим генетический алгоритм построения ИНС. Пусть каждый ген содержит информацию об индексах начального и конечного нейрона в связи, а также вес связи.

Для формирования новых и удаления имеющихся нейронов можно использовать следующие правила:

1. Удаление входного или выходного нейрона не допускается;

2. Новые нейроны получают минимальные из возможных индексов;

3. При удалении нейрона индексы остальных нейронов корректируются с учетом удаленного для предотвращения появления пробелов в индексах.

Для скрещивания используются две особи, производящие двух потомков при помощи двухточечного кроссинговера. Общие нейроны и связи наследуются потомками, а различающиеся элементы наследуются одним из потомков (потомок выбирается случайным образом).

Для мутации введём несколько операторов: добавление нейрона в скрытый слой; удаление случайно выбранного нейрона вместе с его связями; добавление связи между нейронами; удаление связи между нейронами; изменение веса случайно выбранной связи.

За счёт использования скрещивания и мутаций генетический алгоритм способен превзойти метод градиентного спуска. Для полученных в результате потомков рассчитывается значение среднеквадратичной ошибки на выходе сети – приспособленность поколения.

Итак, последовательность шагов применения генетического алгоритма для построения нейронной сети может быть представлена следующим образом:

1. Случайным образом сгенерировать набор нейронных сетей.

2. Оценить приспособленность текущего поколения. Если приспособленность одной или более сетей достаточна, то прекратить выполнение алгоритма. Сеть с наибольшей приспособленностью является результатом работы алгоритма.

3. Сгенерировать новое поколение посредством селекции, скрещивания и мутаций.

При оценке эффективности применения генетического алгоритма в сравнении с применением метода градиентного спуска можно использовать значения таких параметров как среднеквадратичное значение ошибки на нейронах выходного слоя сети и время обучения.

Таким образом, структура и веса связей нейронной сети могут быть сгенерированы с использованием генетического алгоритма. Такой подход позволяет снизить вероятность нахождения неоптимального решения по сравнению с методом градиентного спуска. Нейронная сеть, построенная описанным способом, может быть использована для классификации текстовой информации, поиска литературы по теме и др.

Основным недостатком генетического алгоритма является относительно высокая длительность его работы.

**Список использованных источников:**

1. Бураков, М. В. Генетический алгоритм: теория и практика: учеб. пособие / М. В. Бураков. – СПб. : ГУАП, 2008. – 64 с.

2. Рот, В. И. Система распознавания символов с помощью искусственной нейронной сети / В. И. Рот // XIV Международная научно-практическая конференция студентов, аспирантов и молодых учёных «Молодёжь и современные информационные технологии». – Томск : Изд-во ТПУ, 2016. – С. 54– 55.

3. Метод обратного распространения ошибки [Электронный ресурс]. – Электронные данные. – Режим доступа: [https://www.wikiwand.com/ru/Метод\\_обратного\\_распространения\\_ошибки](https://www.wikiwand.com/ru/Метод_обратного_распространения_ошибки). Дата доступа: 26.03.2020.

4. Генетический алгоритм [Электронный ресурс]. – Электронные данные. – Режим доступа: [http://www.machinelearning.ru/wiki/index.php?title=Генетический\\_алгоритм](http://www.machinelearning.ru/wiki/index.php?title=Генетический_алгоритм). Дата доступа: 26.03.2020.

## ПОДХОД CQRS В WEB-ПРИЛОЖЕНИЯХ НА МОНОЛИТНОЙ АРХИТЕКТУРЕ

Рынкевич А.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Серебряная Л.В. – канд. тех. наук, доцент

Работа посвящена архитектурному приему CQRS, применение которого позволяет создавать гибкие и производительные программные системы. Рассмотрено использование CQRS в web-приложениях на монолитной архитектуре. Показано положительное влияние подхода CQRS на модели программных приложений, а также его удачное сочетание с приемами программирования, улучшающими характеристики различных информационных систем.

Термин «CQRS» «Command and Query Responsibility Segregation» (CQRS) был впервые введен Греггом Янгом и определен им как прием программирования, при котором для совершения операций чтения и записи используются два разных объекта: один – для выполнения команд, другой – для выполнения запросов [1]. Несмотря на всю простоту данного приема, применение CQRS в сочетании с некоторыми другими подходами позволяет сделать систему более производительной и гибкой.

В большинстве архитектур веб-приложений для выполнения операций чтения, записи, изменения или удаления используется одна модель данных. В то время, как это решение отлично применяется в простых системах, в более сложных могут возникнуть следующие проблемы:

1. Формат объектов данных, возвращаемых приложением, сильно отличается от формата, в котором данные хранятся в базе. Время получения таких объектов относительно велико из-за множества операций объединения и большого числа запросов к БД.

2. Требования к скорости выполнения операций чтения и записи отличаются, в то время как модель данных не позволяет удовлетворить их для обеих операций одновременно.

3. Существуют несколько групп пользователей, каждая из которых имеет разные права на получение, добавление, изменение или удаление сущности одного типа. Для создания такого механизма доступа необходима сложная логика, а значит, она больше подвержена ошибкам, что представляет угрозу безопасности хранимой информации.

Хотя CQRS чаще всего используется в системах на микросервисной архитектуре, его можно использовать и в классических монолитных веб-приложениях для решения описанных выше проблем.

Разделим команды и запросы не только на уровне бизнес-логики, но и на уровне базы данных. Схема получившегося приложения в сравнении с классическим подходом представлена на рисунке 1.

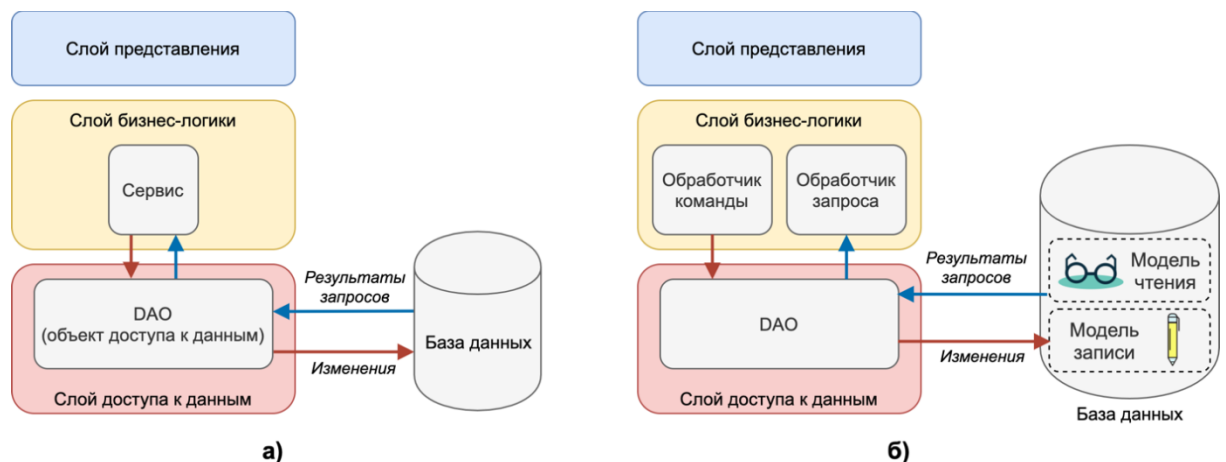


Рисунок 1 – Многослойное веб-приложение без использования CQRS с одной моделью данных (а) и с использованием CQRS и отдельными моделями данных для чтения и записи (б)

Отдельные модели чтения и записи позволяют хранить информацию в том представлении, в

котором это более удобно для каждой из операций. Алгоритм получения информации сводится к отправке простого запроса к хранилищу для получения информации в «готовом» виде.

Стоит отметить, что этот подход предусматривает необходимость синхронизации моделей данных. Однако логика синхронизации может быть вынесена в отдельное фоновое приложение (например, Windows Service) или модуль монолитного веб-приложения с планировщиком задач (Quartz для программ на Java, Quartz.NET для программ на языках платформы .NET или другие аналогичные решения) и выполняться асинхронно.

Асинхронные операции и модель чтения с оптимизированными для представления данными позволяют не только решить проблему снижения производительности из-за отличающихся форматов данных. Они также дают возможность нескольким командам разработчиков параллельно заниматься написанием логики синхронизации и реализацией взаимодействующих с данными модулей.

Развивая идею разделения чтения и записи, можно разместить модели в отдельных базах данных, причем не обязательно одинаковых: например, изменения хранить в реляционной БД, а информацию для чтения – в документной или графовой с подходящими индексами. При выборе способа хранения информации необходимо учитывать характер данных и совершаемые с ними действия, так как от этого зависят средства, которыми будет достигаться уменьшение времени синхронизации и получения информации.

Пример подобного размещения изображен на рисунке 2.

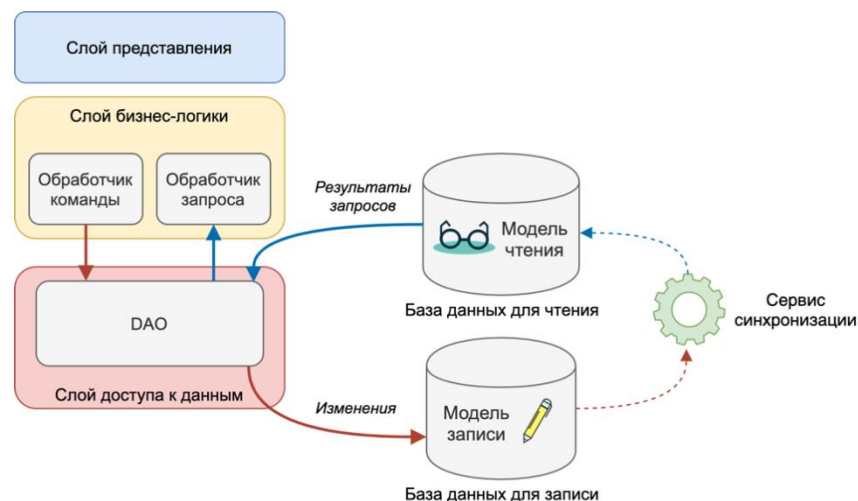


Рисунок 2 – Многослойное веб-приложение с использованием CQRS, отдельными моделями и базами данных для чтения и записи

Адаптируя каждую модель в отдельности, можно удовлетворить требования к скорости выполнения соответствующих операций. Вместе с этим становится проще управление правами и доступом, ведь для каждой сущности выполняется настройка ограничений либо на чтение, либо на запись.

Подход CQRS отлично сочетается с приемом программирования Event Sourcing. Последний предполагает хранение состояния приложения в виде последовательности событий – структур, каждая из которых описывает изменение данных в системе. Для получения актуального состояния системы события «проигрываются» в порядке их поступления. Это позволяет проще обнаруживать и обрабатывать конфликты изменения, а также организовать аудит изменений данных, что может быть важно для администрирования системы. При использовании CQRS модель записи может использоваться как хранилище событий.

Для примера, в банковской информационной системе, использующей Event Sourcing, в качестве событий могут быть представлены транзакции. Тогда суммы на счетах для отображения операционистам и клиентам будут подсчитываться посредством применения событий за сегодняшний день к состоянию счета на вчера, а при закрытии банковского дня, если не наблюдается никаких расхождений в данных, полученные после применения событий состояния счетов будут сохранены в базу.

Использование CQRS можно увидеть во многих архитектурах современных веб-приложений. Одной из них является «Clean Architecture», еще известная как «гексагональная», «луковая» или «порты и адаптеры». Это предметно-ориентированная архитектура, направленная на слабую связность модулей и инверсию зависимостей. В ней CQRS используется для упрощения системы и улучшения ее производительности и масштабируемости.

Таким образом, Command and Query Responsibility Segregation является актуальным архитектурным решением для улучшения характеристик различных информационных систем, в том числе и монолитных веб-приложений.

**Список использованных источников:**

1. CodeBetter; "CQRS, Task Based Uis, Event Sourcing agh!" by Greg Young [Электронный ресурс] – Режим доступа: <http://codebetter.com/gregyoung/2010/02/16/cqrs-task-based-uis-event-sourcing-agh>
2. Martin Fowler: CQRS [Электронный ресурс] – Режим доступа: <https://martinfowler.com/bliki/CQRS.html>
3. CQRS Pattern. Azure Architecture Center | Microsoft Docs [Электронный ресурс] – Режим доступа: <https://docs.microsoft.com/en-us/azure/architecture/patterns/cqrs>
4. E-book "Architect Modern Web Applications with ASP.NET Core and Azure" by Steve "ardalis" Smith [Электронный ресурс] – Режим доступа: <https://aka.ms/MicroservicesEbook>

## **АВТОМАТИЗАЦИЯ СОЗДАНИЯ ИГРОВЫХ ПРИЛОЖЕНИЙ ДЛЯ СОЦИАЛЬНОЙ СЕТИ**

*Савенок И. Л.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Куликов С.С. – канд. тех. наук*

Социальные сети являются одними из важнейших инструментов продвижения продуктов и услуг. Для того, чтобы многократно повысить эффективность продвижения бизнес-аккаунтов в социальных сетях, возможно внедрение геймификации: использование элементов игры в реальной жизни. Для упрощения её реализации применяется автоматизация создания игровых приложений в социальных сетях.

В настоящее время в мире насчитывается свыше 3.8 миллиардов пользователей социальных сетей [1]. Неудивительно, что социальные сети стали оказывать существенное влияние на все сферы жизни общества: социальную, экономическую, политическую, духовную. Социальные сети выступают в качестве одного из основных средств распространения информации, инструментов продвижения продуктов и услуг.

Однако эффективность использования данного инструмента зависит от таких показателей, как уровень вовлеченности (показатель взаимодействия пользователей с публикуемым контентом) и охват (показатель статистики, обозначающий количество пользователей, увидевших определенную информацию сообщества) [2]. Значение охвата, как правило, напрямую зависит от уровня вовлеченности аудитории наряду с рекламой. Поэтому одной из главных задач в повышении эффективности продвижения является увеличение уровня вовлеченности.

Одним из главных трендов 2020 года в продвижении аккаунтов в социальных сетях называют геймификацию: внедрение элементов игры в реальный бизнес [3]. Исследования показывают, что применение геймификации в бизнес-аккаунтах социальной сети Instagram увеличивает вовлеченность подписчиков в несколько раз [4].

Существует большое количество разнообразных способов внедрения геймификации. Более ранние методы подразумевают создание материалов, рассчитанных на одноразовый всплеск активности: конкурсные посты, викторины и загадки в Instagram Stories. Со временем, учитывая трудоемкость создания материалов и их ограниченный срок действия, геймификация в социальных сетях стала нуждаться в автоматизации. Это достигается путем разработки программного средства, которое автоматизирует создание игровых приложений.

В основе такого программного продукта лежит подсчет активности подписчиков аккаунта, а именно сбор количества лайков и комментариев, оставленных каждым пользователем с учетом дополнительных условий:

- количество времени, прошедшего с момента выхода поста до публикации комментария (позволяет мотивировать подписчиков проявлять активность как можно раньше);
- количества слов в комментарии (позволяет учитывать только осмысленные комментарии);
- максимальное количество учитываемых комментариев одного подписчика на посте (позволяет защитить бизнес-аккаунт от спама);
- количество последних постов, на которых учитывается активность подписчиков (позволяет увеличить активность только самых свежих постов).

В последствие за учтенную активность подписчикам присваиваются баллы (игровая валюта), за которую возможно получить подарки либо иные привилегии от владельца бизнес-аккаунта.

Подсчет активности с условий в совокупности с кастомизацией каждого конкурса, а именно указанием условий игры, стоимости различных действий подписчиков и подарков, названия игровой валюты, позволяют разработать многообразные игровые механики.

Таким образом, общее действие программного продукта выражается в быстром создании и настройке приложения для проведения конкурса среди подписчиков, которые, в свою очередь, получают мотивацию проявлять активность в аккаунте создателя, что в последствии ведет к улучшению показателей продвижения и увеличению продаж продуктов и услуг бизнес-аккаунта.

**Список использованных источников:**

1. Digital 2020 Global Digital Overview // datareportal.com [Электронный ресурс]. – Режим доступа: <https://datareportal.com/reports/digital-2020-global-digital-overview>.
2. What is Engagement Rate and Why is it Important in Influencer Marketing? // scrunch.com [Электронный ресурс]. – Режим доступа: <https://www.scrunch.com/blog/influencer-engagement-rates>.
3. Геймификация в Инстаграм: игра за внимание клиента // openbusiness.ru [Электронный ресурс]. – Режим доступа: <https://www.openbusiness.ru/biz/business/gamification-instagram>.
4. Конструктор игр для Instagram // Sub.by [Электронный ресурс]. – Режим доступа: <https://sub.by>.

## ИСПОЛЬЗОВАНИЕ ДИНАМИЧЕСКОЙ СТРУКТУРЫ ИЗ КОНЕЧНЫХ АВТОМАТОВ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ПОИСКА ШАБЛОНА В ТЕКСТЕ

Савёнок В.А., Медведев С.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Медведев С.А. – к.т.н., доцент

Одной из центральных задач в области обработки текстовой информации является задача поиска шаблона в тексте. На сегодняшний день существует множество подходов и алгоритмов для решения данной задачи. Одним из таких подходов является использование конечного автомата. В данной работе представлен оптимальный по трудоемкости подход к построению эффективной по памяти структуры конечных автоматов для решения задачи поиска шаблона в тексте.

Классическим подходом к решению задачи поиска шаблона в тексте является использование конечного автомата. Данный подход хорошо себя зарекомендовал при работе с регулярными и контекстно-независимыми грамматиками [1]. Такие автоматы содержат относительно небольшое число состояний и позволяют сопоставлять достаточно простые шаблоны, в которых применяются такие выражения, как последовательность, вариация и повторение. В то же время, выявление шаблонов, зависящих от контекста, требует проектирования более сложного автомата, число состояний в котором по отношению к сложности шаблонного выражения растет экспоненциально [2].

Для упрощения процесса разработки автомата предложена декомпозиция на простейшие автоматы, объединенные в многоуровневую динамическую структуру. Пример соответствия между деревом выражения поиска шаблона  $\#P = ("Microsoft" \_ "acquires") \dots [0-3] \dots \{ "Google", "Amazon" \}$ ; и структурой автоматов, создаваемой для сопоставления последовательности текстовых лексем, представлен на рисунке 1.

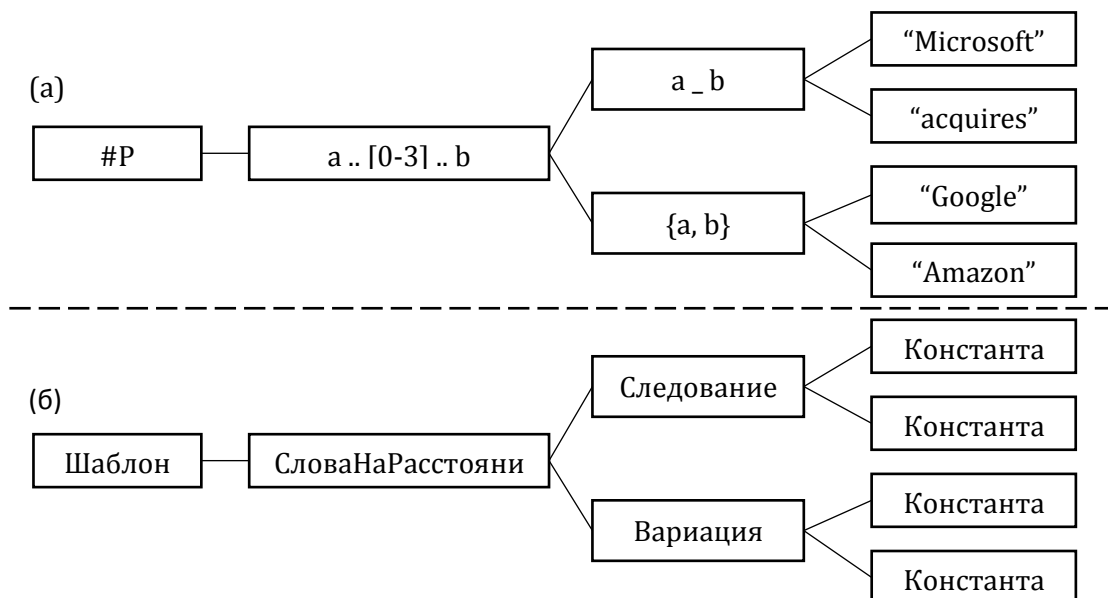


Рисунок 1 – Соответствие между деревом выражения поиска шаблона (а) и структурой автоматов (б)

В разработанной структуре каждый элемент выражения поиска представляется отдельным конечным автоматом с небольшим числом состояний. Такой подход в виде декомпозиции позволяет строить многоуровневые структуры для реализации поиска сложных шаблонов.

Особенность данной структуры заключается в том, что она формируется и модифицируется по мере совпадения отдельных частей выражения шаблона. Автоматы верхнего уровня не будут созданы до тех пор, пока не обработают все автоматы нижнего уровня, что снижает накладные

расходы на проверку шаблонов, которые позже не совпадут. Таким образом повышается эффективность использования памяти и снижается нагрузка на сборщика мусора в языках с автоматическим управлением памятью. В сочетании с событийной моделью взаимодействия автоматов разных уровней, указанный подход позволяет значительно упростить общую структуру поискового движка.

**Список использованных источников:**

1. John E. Hopcroft and Jeffrey D. Ullman Introduction to Automata Theory, Languages, and Computation / John E. Hopcroft [et al.] // Addison-Wesley, 1979. – P. 217.
2. А.С. Морозов Лекции по конечным автоматам и автоматным структурам [Электронный ресурс]. – Режим доступа: <http://math.nsc.ru/~asm256/TA/FANew.pdf>. Дата доступа: 20.03.2020.

## АРХИТЕКТУРНОЕ РЕШЕНИЕ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В СФЕРЕ ТУРИЗМА

Труханович И.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Парамонов А.И. – канд. техн. наук, доцент

В связи с ростом туризма в настоящее время активно развиваются соответствующие системы поддержки принятия решений. В данной работе предложен вариант архитектурного решения такой системы, которая по предпочтениям и ограничениям потребителей подбирает оптимизированные туристические маршруты.

В настоящий момент перед классом систем поддержки принятия решений в сфере туризма ставится различного рода цели по осуществлению помощи в туристической деятельности [1].

В рамках данной работы разрабатывается информационная система поддержки турбизнеса. Основные функциональные задачи, которые ставятся перед данной системой, можно сформулировать так: построение списка маршрутов на основе данных о предпочтениях потребительской аудитории, динамический отбор и модернизация маршрутов на основе ограничений клиента по времени и затратам. В качестве входных данных системы выступает модель клиента, которая включает такие основные параметры: возраст, темперамент, предпочтения и ограничения

Было рассмотрено несколько вариантов технологических решений разрабатываемой системы. В качестве одного из вариантов предложено архитектурное решение, в основе которого предполагается использовать конвейер обработки данных. Например, для этого можно использовать инфраструктуру Amazon Web Services [2]. Данное облако хорошо зарекомендовало себя в качестве надёжного и масштабируемого сервиса. Предлагаемый вариант системы представлен на рисунке 1.

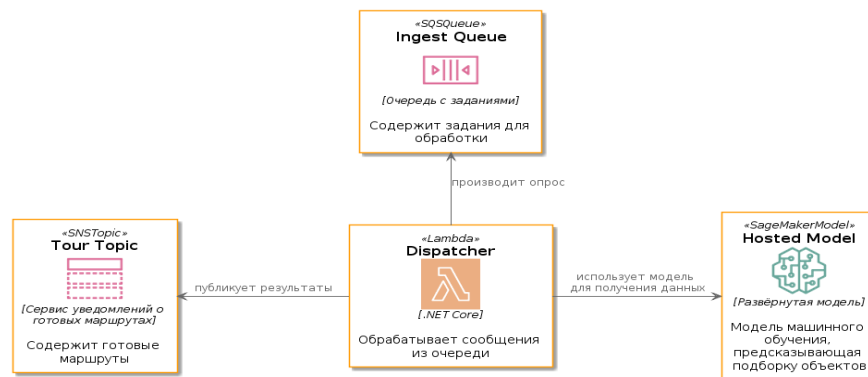


Рисунок 1 – Архитектурное решение на основе конвейера обработки данных

Все входные данные в виде модели с параметрами пользователей поступают на очередь с заданиями (Ingest Queue). Обработчик (Dispatcher) опрашивает очередь входных данных, доставая из неё задания для обработки, после чего отправляет эти данные на развернутую модель машинного обучения (Hosted Model). Hosted Model реализует задачу подбора объектов и возвращает обработчику список предполагаемых туристических объектов. На основе этого списка объектов формируется итоговый вариант маршрута путём применения ряда оптимизаций. Сначала с помощью алгоритма “укладки рюкзака” формируются варианты “туристических пакетов”, в которые входят доступные по ценовой категории объекты без учета навигации между ними [3]. Затем алгоритм задачи коммивояжера применяется многократно к различным “туристическим пакетам” для



построения кратчайшего пути из множества доступных клиенту объектов в этом пакете с учетом ограничения по времени маршрута. Подобранный маршрут преобразуется в JSON формат и передается на сервис уведомлений (Tour Topic), который предоставляет результаты работы системы конечному потребителю.

Реализация обработчика выполнена на платформе .NET Core.

**Список использованных источников:**

1. Benckendorff P.J. Tourism Information Technology / P.J. Benckendorff, Z. Xiang, P.J. Sheldon // Wallingford: CABI, 2019. – 401 p.
2. About AWS [Electronic resource] : AWS. – Mode of access: <https://aws.amazon.com/about-aws/>. – Date of access: 20.03.2020.
3. Парамонов, А.И. Подходы к решению задачи автоматизированного составления экскурсионных туров / А.И. Парамонов, И.А. Труханович // Information Technologies and Systems: материалы международной научной конференции / Белорусский государственный университет информатики и радиоэлектроники; редкол. : Л. Ю. Шилин [и др.]. – Минск, 2019. – С. 74 – 75.

## ОБРАБОТКА БЛАНКОВ ОТВЕТОВ ДЛЯ АВТОМАТИЗАЦИИ ПРОВЕДЕНИЯ БУМАЖНОГО ТЕСТИРОВАНИЯ

*Харлов А.А., Парамонов А.И.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Парамонов А.И. – канд. техн. наук, доцент*

В работе описаны этапы обработки изображений подготовленных бланков для проведения бумажного тестирования. Представлены результаты проведенных экспериментов по распознаванию изображений бумажных бланков.

Рассматриваемый в работе вид бумажного тестирования предполагает ответы испытуемого на заранее подготовленном форматном бланке, для подготовки которого зачастую используется специальное программное обеспечение [1]. Пример такого бланка показан на рисунке 1. После заполнения бланка ответов экзаменуемый сдает его экзаменатору, который сканирует бланк и загружает изображение в систему проверки.



Рисунок 1 – Заполненный бланк ответов бумажного тестирования

Обработка изображений бланков бумажного тестирования включает в себя несколько этапов:

1. распознавание формы бланка на изображении;
2. выравнивание изображения по осям x, y;
3. распознавание QR-кода;
4. распознавание ответов.

Выравнивание изображения необходимо поскольку оно может быть повернуто при сканировании, как это показано на рисунке 1. Выравнивание осуществляется с помощью QR-кода, который содержит в себе выравнивающий узор. Далее происходит извлечение данных о тесте, которые зашиты в QR-коде. Для нахождения и распознавания QR-кода используется библиотека ZXing.Net [2]. Распознав QR-код и зная его координаты, рассчитывается местоположение таблицы ответов. Предполагается, что она находится на одной линии с QR-кодом. Результат выделения таблицы с ответами показан на рисунке 2. Для поиска объектов на изображении используется Accord.Net [3].

	1	2	3	4	5	6	7	8	9	10
a	X					X				
b			V					+		
c		V								
d							+			
e									●	

Рисунок 2 – Изображение таблицы бланка ответов

По полученному объекту таблицы далее определяются границы каждой ячейки, а внутри каждой ячейки определяется процент заполнения области. Для повышения качества распознавания ячеек применяются фильтры для изображений из библиотеки Accord.Net, а именно:

- Grayscale (перевод изображения в черно-белое);
- BradleyLocalThresholding (алгоритм считает, что каждый пиксель изображения имеет черный цвет, если его яркость на  $t$  процентов ниже, чем средняя яркость окружающих пикселей в окне указанного размера, в противном случае устанавливается белый цвет);
- HomogeneityEdgeDetector (алгоритм находит края объектов, вычисляя максимальную разницу пикселей обработки с соседними пикселями в 8 направлениях);
- Invert (инвертирование цвета изображения).

Все изображения, которые загружаются в систему для обработки, сохраняются в базе данных программного обеспечения, что позволяет использовать их для уточнения полученной оценки.

После успешного распознавания изображения бланка программное средство отобразит в диалоговом окне сетку распознанных ответов с пометкой о правильности ответа, полученной оценкой и кнопкой «Подтвердить оценку» (см. рисунок 3).

Результат распознавания ×

	1	2	3	4	5	6	7	8	9	10
a	x					x				
b			x					x		
c		x			x					
d				x			x			
e									x	
	×	×	×	×	✓	×	✓	×	×	×

Правильных ответов: 2 / 10 Оценка: 2

Отмена
Подтвердить оценку

Рисунок 3 – Диалоговое окно результата обработки бланка

Перед подтверждением оценки в программе экзаменатор имеет возможность при необходимости исправить ошибки распознавания изображения бланка (таблица ответов является редактируемой). Подтверждение оценки фиксируется в базе данных. После чего оценка может быть использована для построения статистики по экзаменуемым, либо для заполнения табеля аттестации при условии ввода/выбора ФИО испытуемого (см. рисунок 4).

Поскольку точность распознавания зависит от исходного изображения, полученного при сканировании бланка ответов, был проведен эксперимент для выяснения минимально допустимого качества изображения. В результате ряда экспериментов установлено, что параметр разрешения изображения при сканировании должен быть не менее 200 dpi.

В ходе экспериментов были рассмотрены варианты получения изображения для обработки с распознаванием фотографии бланка с телефона. На данный момент программное средство показывает низкую точность распознавания бланков с фото, так как изображение, полученное на камеру телефона, получается неоднородным по цвету, а также, в отличие от изображения со сканера, имеет искажения в наклоне. Полученные результаты были оформлены в виде дополнительных технических требований, которые необходимо учитывать при разработке программных средств по обработке бумажных бланков.

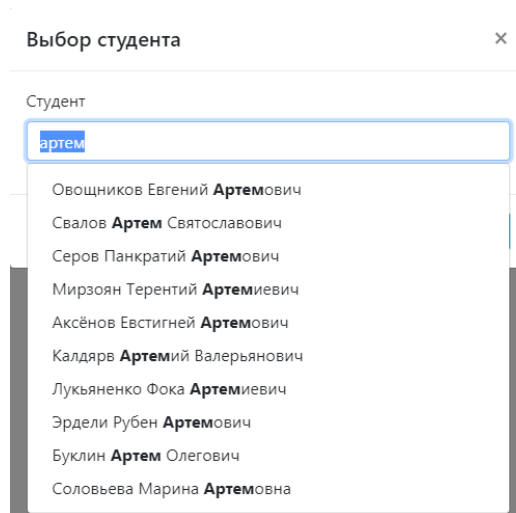


Рисунок 4 – Диалоговое окно выбора экзаменуемого

**Список использованных источников:**

1. Парамонов А.И. Программное обеспечение для автоматизации проведения бумажного тестирования / А.И. Парамонов, А.А. Харлов // Вычислительные методы, модели и образовательные технологии: сб. материалов VII междунар. НПК, Брест, 18.10.201; под общ. ред. А.А.Козинского. – Брест: БрГУ, 2019. – С. 188-190.
2. ZXing.Net [Электронный ресурс]. – Режим доступа: <https://github.com/micjahn/ZXing.Net/blob/master/README.md>. Дата доступа: 22.03.2020.
3. Accord.Net [Электронный ресурс]. – Режим доступа: <http://accord-framework.net/>. Дата доступа: 22.03.2020.

## МЕТОДЫ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ ДЛЯ ВЫЯВЛЕНИЯ АНОМАЛИЙ В ПОСЛЕПОЛЕТНЫХ ДАННЫХ ОБЪЕКТИВНОГО КОНТРОЛЯ БПЛА

*Хотько А.К., Вербицкий В.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Кулага В.В. – канд. техн. наук, доцент*

Развитие информационных технологий, методов записи и хранения данных привело к росту объемов собираемой информации. Часто среди собранных данных находятся информация, которая может оказаться крайне полезной для решения определенных задач, таких как обнаружение неполадок в работе устройств, прогнозирования погоды, распознавания спама и мошенничества и др. Объемы собираемых данных увеличились до таких размеров, что человеку просто не по силам проанализировать их вручную. Для того чтобы автоматизировать процесс анализа данных используются различные подходы, одним из которых является интеллектуальный анализ данных (англ. Data mining).

Одним из основных направлений технической эксплуатации авиационной техники является повышение надежности системы управления полетом. В гражданской и военной авиации обработка полетных данных играет важную роль в деле повышения безопасности полетов. Оценить работоспособность систем в полете возможно с помощью телеметрической информации, поступающей с устройств и датчиков систем. Авиационными правилами Республики Беларусь предусмотрена обязательная процедура анализа полетных данных предыдущих полетов перед выполнением новых полетов [1]. Очень часто выполнение подобных работ требует оперативности, минимизации времени, затрачиваемого на анализ данных полета. В связи с этим возникает необходимость в автоматизации процесса анализа послеполетных данных на предмет аномалий.

Аномалия – это отклонение поведения системы от стандартного, ожидаемого. В различных литературных источниках аномалия также упоминается как выброс, ошибка, отклонение или исключение. Анализ потока данных на обнаружение отклонений является одной из задач интеллектуального анализа данных.

Существует множество различных вариантов классификации существующих методов поиска аномалий. Были рассмотрены следующие группы методов обнаружения аномалий [2]:

- классификация;
- статистический анализ;
- алгоритм ближайшего соседа.

В результате аналитического обзора вышеперечисленных групп методов было определено, что

оптимальным методом обнаружения аномалий для использования в обработке данных объективного контроля полета беспилотного летательного аппарата является метод «опорных векторов» из группы методов «классификация». Данный метод был выбран исходя из того, что возможные аномалии во время исследования послеполетных данных, зафиксированных с помощью датчиков летательного аппарата в течение полета, будут заданы некоторыми границами критических значений параметров, что позволит однозначно, быстро и достаточно точно определять аномалии среди значений параметров, полученных с датчиков.

Возможно, метод статистического анализа в некоторых случаях дал бы более точные результаты анализа данных, однако построение достоверной модели поведения БПЛА является достаточно трудоемким процессом. Принимая во внимание сложность процессов, происходящих с беспилотным летательным аппаратом во время полета, становится очевидно, что такая модель будет зависеть от значительного числа параметров, такие как скорость и направление ветра, высота полета, атмосферное давление и др.

**Список использованных источников:**

1. Об утверждении *Авиационных правил «Выдача свидетельств и классификация авиационного персонала гражданской авиации Республики Беларусь»* [Электронный ресурс] : Постановление Министерства транспорта и коммуникаций Республики Беларусь, 19 апреля 2018 г., № 14 // Национальный правовой Интернет-портал Республики Беларусь. – Режим доступа: [http://pravo.by/upload/docs/op/W21833443p\\_1538773200.pdf](http://pravo.by/upload/docs/op/W21833443p_1538773200.pdf). – Дата доступа: 25.03.2020.

2. Обзор методов обнаружения аномалий в потоках данных : материалы II конф. по программной инженерии и организации информации, Санкт-Петербурга, 21 апреля 2017 г. / Санкт-Петербургский гос. ун-т ; редкол.: Г. А. Чернышев (гл. ред.) [и др.]. – СПб., : СПбГУ, 2017. – 56 с.

## **ИНФОРМАЦИОННАЯ СИСТЕМА СБОРА И ОБРАБОТКИ ТЕЛЕМЕТРИЧЕСКИХ ДАННЫХ**

*Малец И.В., Тимофеев Н.И., Моженкова Е.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Парамонов А.И. – к.т.н., доцент*

В современном мире, наполненном большим количеством информационных систем и технологий, важное место занимает их всесторонний мониторинг. Суть его заключается в сборе необходимых данных и их анализе. Анализ собранной информации об устройстве или программном средстве позволяет предотвратить ошибки или, на основании полученной информации, оптимизировать их производительность. Практически все современные устройства и программные средства имеют собственные системы мониторинга. Но в случае, когда количество систем для мониторинга слишком велико, или к ним нет непосредственного доступа, – стандартных средств становится недостаточно.

Сбор, хранение и обработка телеметрических данных становятся все более актуальными задачами. На сегодня известно уже большое количество систем и программных средств для её решения. Часть из них нацелена на мониторинг конкретных типов систем – программные средства для мониторинга сетей: *Observium* [1], *SolarWinds Network Performance Monitor* [2]. Другие, более гибкие, могут работать с большим количеством устройств и протоколов, по средствам добавления в них модулей или установки специальных приложений-агентов. Примером таких систем являются *Zabbix* [3] или *CA UIM* [4]. Практически все современные системы сбора и обработки телеметрических данных имеют веб-интерфейс для доступа к данным из любой точки мира. В последнее время многие системы для сбора и обработки телеметрических данных становятся доступны в формате облачных сервисов, что упрощает для пользователей настройку, увеличивает надежность и отказоустойчивость системы и не требует от пользователя мощного оборудования и больших объемов памяти.

Существует два основных способа сбора телеметрических данных. Первый способ – система для сбора телеметрических данных сама опрашивает устройства или программное обеспечение по средствам протокола или программного интерфейса. Второй способ – приложения-агенты, которые опрашивают устройство или программное обеспечение и передают данные на сервер системы мониторинга. Приложения агенты так же, как и в первом случае, опрашивают устройства или программные средства с помощью протокола или программного интерфейса, либо же являются частью опрашиваемых систем. В одной системе одновременно могут использоваться оба способа. Системы использующие первый способ обычно более просты в конфигурации, но сложны в доработке. Второй способ может потребовать установки приложения-агента непосредственно на целевое устройство, а также настройки самой системы и приложений агентов по отдельности для каждого источника телеметрических данных.

Преимущества второго способа – это гибкость, возможность разработки новых приложений агентов и модификации существующих, независимо от основных частей системы и других приложений агентов.

Пример систем с приложениями агентами и без них представлены на рисунке 1.

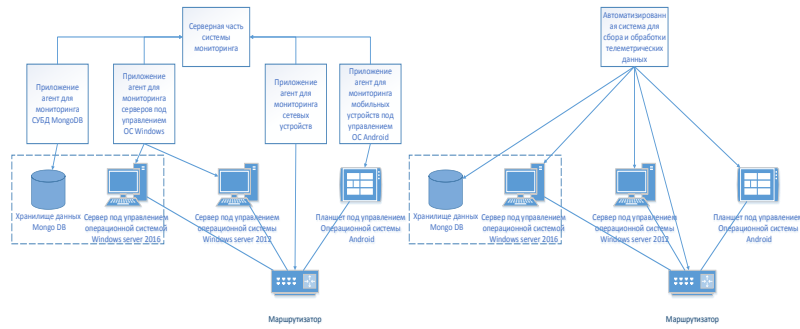


Рисунок 1 – Пример систем с приложениями агентами и без агентов

Анализ систем сбора и обработки телеметрических данных показал, что большинство существующих систем разрабатываются для конкретного сценария использования и типа пользователя (например, для администраторов сетей) и сложны в настройке. Поэтому была предпринята попытка разработать простую в настройке и использовании, гибкую систему для сбора и обработки телеметрических данных.

На рисунке 2 представлена страница веб интерфейса разработанного приложения с информацией об источнике данных (iPhone SE). На странице изображен блок с названием устройства, типом и статусом. Можно просмотреть данные устройства в реальном времени или за выбранный период, установить пороговые значения для данных.

Система имеет серверную часть, которая принимает, хранит и обрабатывает данные, простой веб-интерфейс с возможностью просмотра и анализа данных разного типа, получения данных по средствам приложений агентов. Система имеет открытый исходный код и открытый API. Открытый исходный код позволит конечным пользователям, при необходимости, доработать систему для своих нужд. Открытый API позволит пользователям или третьим лицам разрабатывать новых агентов для системы. На начальных этапах работы системы разработано несколько приложений-агентов для мониторинга наиболее популярных систем и протоколов (для персональных компьютеров, работающих на базе операционных систем Windows, Linux. Для SNMP протокола). Одним из первых разработано приложение-агент для мониторинга мобильных устройств. Несмотря на популярность такого типа устройств, приложения для их мониторинга практически отсутствуют.

Использование системы мониторинга позволит упростить получение детальной информации о различных системах за счет снижения количества рутинных операций.

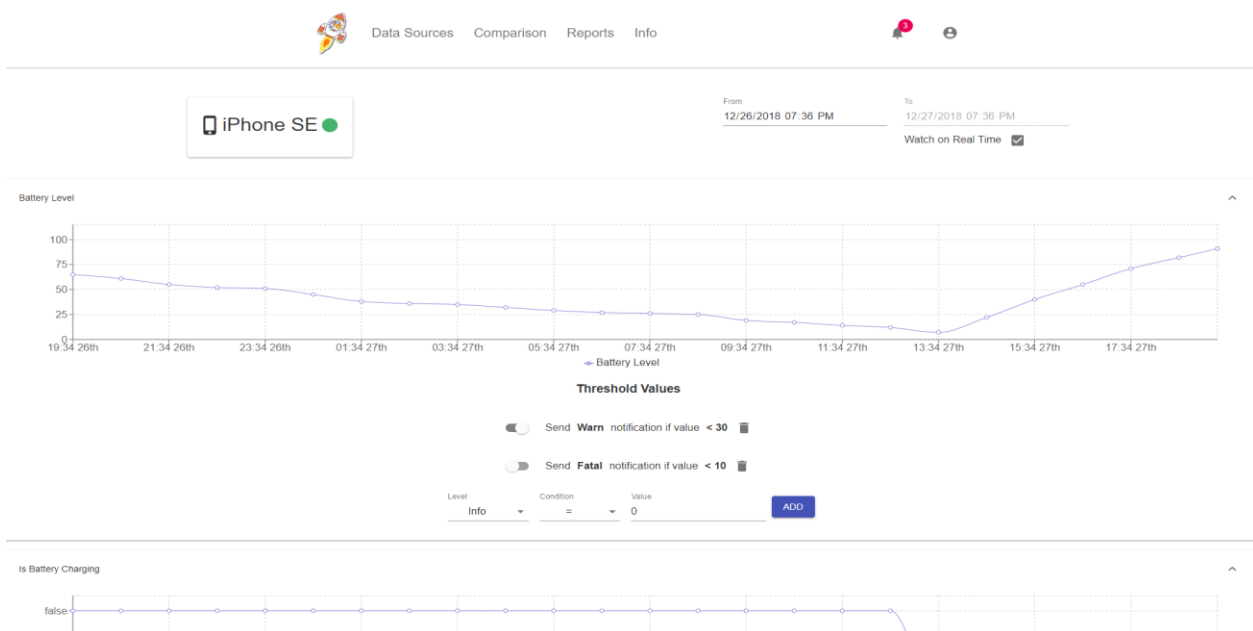


Рисунок 2 – Страница с данными устройства

Планируется распространять продукт в открытом доступе с открытым исходным кодом, а также при необходимости дорабатывать систему под требования заказчика. Такая система может быть

использована системными администраторами для мониторинга сетей, пользователями с большим количеством устройств. Для мониторинга умных домов и многого другого, за счет разработки новых приложений агентов.

**Список использованных источников**

1. Observium [Электронный ресурс] – Электронные данные. – Режим доступа: <http://www.observium.org>. Дата доступа: 12.02.2019 г.
2. Solarwinds Network Performance Monitor [Электронный ресурс] – Электронные данные. – Режим доступа: <https://www.solarwinds.com/network-performance-monitor>. Дата доступа: 12.02.2019 г.
3. Zabbix [Электронный ресурс] – Электронные данные. – Режим доступа: <https://www.zabbix.com/ru/>. Дата доступа: 12.02.2019 г.
4. CA Unified Infrastructure Management (UIM) [Электронный ресурс] – Электронные данные. – Режим доступа: <https://www.ca.com/us/products/ca-unified-infrastructure-management.html>. Дата доступа: 12.02.2019 г.

## ОБЩИЕ ПОДХОДЫ К ПОСТРОЕНИЮ ЭМУЛЯТОРОВ X86-СОВМЕСТИМЫХ ПРОЦЕССОРОВ

*Шпаковский А.П.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Оношко Д.Е. – старший преподаватель*

В настоящее время не теряет актуальности построение эмуляторов аппаратно-программных платформ, применение которых, с учетом современных реалий, проблематично или нецелесообразно из соображений, как правило, безопасности и удобства работы с памятью.

Одним из ключевых направлений деятельности разработчиков современных эмуляторов является поддержка процессоров архитектуры x86, работающих в реальном режиме[1]. Несмотря на то, что этот режим давно перестал быть основным режимом на современных процессорах, не стоит забывать о его предельной простоте, приходящейся весьма кстати для образовательных целей и знакомства с x86-архитектурой.

На момент разработки проекта можно выделить несколько программных средств, предоставляющих подобные возможности. Наиболее популярные из них - эмуляторы Vochs[3] и DOSBox[2], в центре внимания которых эмуляция процессоров архитектуры x86, в частности, их реального режима. В обоих случаях разработчики своей целью ставили поддержание максимально большого количества ПО для разных моделей процессоров и периферийных устройств. Обеспечение подобного рода гибкости в той или иной мере негативно сказывается на скорости работы программы ввиду невозможности применения одинаковых способов эмуляции и возможных оптимизаций для разных модификаций эмулируемого оборудования.

В разработанном прототипе эмулятора за основу взяты характеристики процессора Intel 8086, положившего начало архитектуре x86. Целевая платформа также использует обратно совместимую с x86 архитектуру и набор команд, что позволяет значительно оптимизировать некоторую алгоритмически реализуемую часть задач, переложив ее на реальное «железо».

Рассмотрим более подробно реализацию некоторых узлов системы и приемы эмуляции.

**Регистры.** В качестве эмулируемых регистров удобно использовать 16-битные участки памяти, над которыми выполняются те или иные операции.

**Флаги.** Как уже было сказано, ввиду схожести архитектуры гостевой и хостовой платформы и семантической схожести эмулируемых и аппаратно выполняющихся инструкций, допустимо использование некоторых флагов реального процессора. Достаточно считать их из регистра флагов после выполнения очередной инструкции до того, как они изменятся.

Достигается это копированием содержимого регистра флагов с помощью инструкций PUSHFD и POP. Первая инструкция помещает флаги на стек, откуда, с помощью второй команды, они переносятся в выбранный регистр или область памяти, где ими можно манипулировать с помощью битовых операций.

**Разбор инструкций.** Первостепенной задачей эмулятора, без которой не имеет смысла все вышеописанное, является разбор и выполнение инструкций, записанных во входном файле. Наиболее простым и удобным вариантом является применение массива указателей на процедуры, которыми эмулируются те или иные машинные команды. В документации Intel[4, Vol. 2C A-8] его принято представлять в виде матрицы 16×16. Необходимый указатель находится в строке с номером, соответствующим старшей тетраде первого байта текущей инструкции. Младшая тетрада задает, соответственно, положение в этой строке.

Самыми распространенными из эмулируемого набора инструкций являются команды,

которые, согласно современной документации Intel [4], принято называть инструкциями с ModR/M байтом. Для этого случая были разработаны несколько процедур, позволяющие разбирать такие инструкции одинаково, вне зависимости от типа операндов, ведь на хостовой системе и то, и другое является оперативной памятью.

Работает это так: исходя из того, что размер и тип операндов, которые может принимать инструкция известен заранее, вызывается одна из четырех возможных функций, работающих со своим «шаблоном». В теле функции на основании структуры ModR/M байта информация об операндах конкретизируется и, основываясь на уже точных данных, вычисляются адреса эмулируемых регистров или операндов в памяти. Эти вычисления, для удобства применения, также были вынесены в отдельные подпрограммы.

**Прерывания.** Не менее важной задачей является поддержка прерываний и, соответственно, функций гостевой системы. Здесь проблематично применить унифицированный подход для эмуляции всех прерываний, так как эмуляция реального режима производится на защищенном. Соответственно набор инструментов, к которым можно получить прямой доступ существенно ограничен и эмуляция некоторых аппаратных узлов достигается взаимодействием с интерфейсом хостовой ОС.

Как частный случай рассмотрим эмуляцию функции получения символа из стандартного потока ввода, реализуемую на ОС Windows. Дойдя до вызова этой функции цикл разбора и выполнения инструкций прерывается, а адрес ожидающей выполнения инструкции сохраняется. От окна ожидается сообщение WM\_CHAR [5]. После его получения в регистр помещается код считанного символа и цикл разбора инструкций запускается заново, имея на входе адрес инструкции, до которой еще не дошло выполнение.

Таким образом, в результате проделанной работы был реализован набор приемов эмуляции основных узлов процессора архитектуры x86, достаточный для выполнения некоторых программ, подразумевающих работу в реальном режиме. Весь набор реализованных инструментов удалось разбить на логически обособленные подпрограммы. Получившиеся «блоки» можно применять отдельно от остальных или изменять один без внесения правок в остальные, а также добавлять новые, расширяя возможности эмулятора.

**Список использованных источников:**

1. Real mode – Wikipedia [Электронный ресурс]. - Режим доступа: [https://en.wikipedia.org/wiki/Real\\_mode](https://en.wikipedia.org/wiki/Real_mode). Дата доступа: 10.12.2019
2. DOSBox v0.74-3 Manual [Электронный ресурс].—Режим доступа: <https://www.dosbox.com/DOSBoxManual.html> - Дата доступа: 06.02.2020 г.
3. Bochs Developers Guide [Электронный ресурс].—Режим доступа: <http://bochs.sourceforge.net/doc/docbook/development/index.html> - Дата доступа: 04.02.2020 г.
4. Intel 64 and IA-32 Architectures Software Developer's Manual / Intel Corporation, 2014
5. WM\_CHAR message - Microsoft Software Developer's Network [Электронный ресурс].—Режим доступа: <https://docs.microsoft.com/en-us/windows/win32/inputdev/wm-char> — Дата доступа: 12.12.2019 г.

## **МЕТОДЫ ЗАЩИТЫ АВТОРСКОГО ПРАВА НА ПРОГРАММНЫЕ ПРОДУКТЫ С ПОМОЩЬЮ ВОДЯНЫХ ЗНАКОВ И ОТПЕЧАТКОВ ПАЛЬЦЕВ**

*Шулицкий Д.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Ярмолик В.Н. – д.т.н., профессор*

Защита цифровых работ от нелегального использования и распространения является сложной задачей в информационном обществе. Полное предотвращение незаконного использования работ недостижимо за разумную цену. Большинство схем защиты авторского права отпугивают людей от нелегального использования и распространения цифрового контента тем, что позволяют обнаружить незаконное использование. Для этого в оригинальную работу добавляют идентификационную информацию с помощью техники водяных знаков или отпечатков пальцев.

Цифровой водяной знак – это информация, внедрённая в цифровую работу, которая позволяет автору работы доказать своё авторство. В самом простом случае водяной знак представляет собой строку, указывающую, что данная работа охраняется авторским правом. Цифровой отпечаток пальца является водяным знаком, в котором содержится не только информация об авторе, но и о субъекте, которому предоставлено право пользования произведением [1].

Если функция извлечения водяного знака общедоступна, водяной знак называют видимым. Если эта функция доступна только лицу, внедрившему цифровой водяной знак, его называют невидимым.

Водяные знаки в программном обеспечении можно разделить на водяные знаки данных и водяные знаки кода, а также на статические и динамические. Водяные знаки данных размещаются в неиспользуемых структурах данных программы. Водяные знаки кода внедряются при помощи манипуляций с инструкциями микропроцессора. Статические водяные знаки извлекаются непосредственно из файла программы. Для извлечения динамических водяных знаков необходим запуск программы и получение результата её работы [2].

Цифровые отпечатки пальцев, в отличие от обычных водяных знаков, содержат не только информацию об авторе программы, но и информацию о покупателе программы. При продаже очередной цифровой копии программы формируется содержимое отпечатка пальца. Далее эта информация зашифровывается при помощи приватного ключа разработчика. Это позволяет доказать, что цифровой отпечаток пальца был внедрён конкретным лицом. Затем отпечаток пальца внедряется в программное обеспечение. Отпечатки пальца позволяют обнаруживать источники нелегального распространения программы, так как не составляет труда определить покупателя конкретной копии программы.

К цифровым отпечаткам пальцев предъявляются дополнительные требования по сравнению с водяными знаками: даже если атакующий имеет доступ к некоторому количеству копий программы, он не должен иметь возможность прочесть, повредить или удалить отпечаток пальца на основе информации, полученной путём сравнения копий.

Концептуальной проблемой обычных водяных знаков является то, что демонстрация присутствия водяного знака как свидетельства раскрывает чувствительную информацию, которая может быть использована для удаления водяного знака. Желательно убедить верификатора в том, что водяной знак присутствует и не раскрыть ему информацию, которая может помочь удалить водяной знак. Одним из подходов к решению данной проблемы является использование слепой проверки водяных знаков. Слепые протоколы позволяют убедить верификатора, что правообладатель знает секретное значение, и верификатор не узнаёт ничего нового о секретных данных правообладателя. Например, правообладатель может создать настоящий водяной знак и спрятать его в большом списке поддельных водяных знаков. Затем он предлагает верификатору обнаружить все водяные знаки и доказывает, что, по крайней мере, один из водяных знаков является настоящим, не раскрывая, какой именно. Безопасность данного метода основана на том, что количество водяных знаков в списке должно быть на столько большим, что невозможно удалить их все без серьёзного ухудшения стего-данных [3].

**Список использованных источников:**

1. Криптография, стеганография и охрана авторского права / В.Н.Ярмолик, С.С. Портянко, С.В.Ярмолик / Издательский центр БГУ — Минск, 2007. — С. 195–211
2. Watermarking, Tamper-Proofing and Obfuscation – Tools for Software Protection / C. Collberg, C. Thomborson – Department of Computer Science University of Arizona, 2000. – P. 7-11.
3. Zero-Knowledge Watermark Detection and Proof of Ownership / A. Adelsbach, A. Sadeghi – LNCS, 2001. – P. 273-288.

## **ПРОГРАММНОЕ СРЕДСТВО АНАЛИЗА РЕЛЬЕФА МЕСТНОСТИ НА ОСНОВЕ ДАННЫХ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ**

*Юхневич Ю.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Шамына А.Ю. – м.т.н., асс. каф. ПОИТ*

В данной работе описаны некоторые стороны разработки программного средства для анализа рельефа местности на основе данных дистанционного зондирования Земли, позволяющее загружать и обрабатывать соответствующие данные, а также получать профиль местности.

Основным назначением разработки данного программного средства является построение профиля местности, а также немаловажным является предоставление удобного доступа к данной функциональности, не только для приложения, реализованного в рамках данного программного средства, но и для других систем.

Профили местности используются для решения большого числа различных задач: при составлении проектов строительства инженерных сооружений, при составлении геологических разрезов, при определении линий видимости между выбранными точками, для составления описания рельефа по маршруту, для расчёта загрязнения атмосферы и др.

В качестве исходных данных для работы программного средства использовались данные дистанционного зондирования Земли, полученные в результате Shuttle Radar Topography Mission



(SRTM).

Shuttle Radar Topography Mission (SRTM) [1] – международная миссия по получению данных цифровой модели рельефа Земли. Сбор данных производился 11 – 22 февраля 2000 года. В этом международном проекте также участвовали NASA и Национальное агентство геопространственной разведки (NGA). Съёмка осуществлялась с борта шаттла «Идевор» с помощью радара SIR-C/X-SAR. Собранные радиолокационные данные были использованы для создания первого почти глобального набора высотных данных.

SRTM успешно исследовала около 80% земной поверхности между 60° северной и 56° южной широтами. Для территории США доступны данные с размером ячейки 1 угловая секунда (пространственное разрешение 30 м), для остальной поверхности Земли – с размером ячейки 3 угловые секунды (пространственное разрешение 90 м). Расчёты высот выполнялись относительно эллипсоида WGS84.

Исходные данные распространяются бесплатно и доступны в различных форматах и версиях.

Существуют три версии данных: предварительная, окончательная и обработанная. Окончательная версия прошла дополнительную обработку выделение береговых линий и водных объектов, фильтрацию ошибочных значений. Обработанная версия производится CGIAR (Consultative Group for International Agricultural Research) и представляет собой сборку мозаик в более крупные фрагменты (5x5 градусов) и исправление областей с отсутствующими значениями.

В работе программного средства используются файлы GeoTIFF формата. GeoTIFF – это TIFF файл, содержащий географическую информацию. Данный формат обычно используется для изображений в геоинформационных системах.

Хранение данных организовано с использованием MS SQL сервера, так как он имеет встроенную поддержку работы с географическими данными.

Для импорта данных из GeoTIFF файлов было разработано отдельное приложение. Данное приложение представляет собой консольную программу, которая получает данные о географических точках из изображения и записывает их в базу данных. Так как для некоторых точек исходные данные не имеют определённого значения, т.е. имеют значение по data (-32768), то в конечном наборе имеются пропуски.

Основное программное средство предоставляет веб-сервис для доступа к пространственным данным. Веб-сервис реализован в виде точки доступа WebAPI. Необходимыми параметрами для получения профиля местности являются: географическая точка начала, направление, расстояние и шаг определения высоты. Результатом запроса является массив данных, определяющий профиль местности.

Для наглядной демонстрации работы программного средства было разработано веб-приложение, взаимодействующее непосредственно с вышеописанным веб-сервисом.

**Список использованных источников:**

1. USGS EROS Archive - Digital Elevation - Shuttle Radar Topography Mission (SRTM) Non-Void Filled [Electronic resource]. – Mode of access: <https://www.usgs.gov/centers/eros/science/usgs-eros-archive-digital-elevation-shuttle-radar-topography-mission-srtm-non>. – Date of access: 27.03.2020.

## **АНАЛИТИЧЕСКАЯ МОДЕЛЬ ТЕЧЕНИЯ ОНКОЛОГИЧЕСКИХ ЗАБОЛЕВАНИЙ**

*Столбовская П.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Шамына А.Ю. – магистр техн. наук, ассистент каф. ПОИТ*

В работе рассматривается возможность построения аналитической модели течения онкологических заболеваний на основе статических данных. Описан подход по выбору наиболее значимых параметров из предложенных. Методом простой линейной регрессии построена математическая модель продолжительности жизни после операции. Анализируется возможность использования полученной модели при решении научно-исследовательских и прикладных задач

Актуальность вопроса нахождения оптимальных путей лечения онкологических заболеваний сохранилась по сей день. В настоящее время решения проблем, связанных с онкологическими заболеваниями, имеют особую значимость. По данным ВОЗ в 2014 году в Беларуси количество смертей, главной причиной которых является онкология, составило 122 000 человек. Рак лёгкого, трахеи или бронх среди мужчин – 26,1% [1].

Данные по раку легких за последние 15 лет, предоставленные Минским городским клиническим онкологическим диспансером, содержат упорядоченные наборы характеристик организма пациента,

имеющихся с рождения или же приобретенных в результате воздействия на организмы применяемых методов, таких как химиотерапия или лучевая терапия.

Предоставленные данные также содержат сведения о генетических особенностях человеческого организма, наличие и описание привычек, сроки наблюдения и его условия, диагноз и проведенное лечение. В связи с невозможностью однозначной оценки каждого пациента некоторые поля данных отсутствуют или не соответствуют общему шаблону. Модель позволяет отобразить значительность факторов наглядно, а также может применяться при построении методов и алгоритмов обработки данных в программном средстве учета и анализа информации о течении онкологических заболеваний.

Из анализируемых данных была сформирована выборка по пациентам с неблагоприятным исходом лечения. Для каждого пациента из полученной выборки вычислена продолжительность жизни после операции. С целью оценки влияния той или иной характеристики на продолжительность жизни были вычислены коэффициенты корреляции Пирсона для следующих факторов:

Фактор курения (S) – 0,0861.

T (tumor) – 0,29867.

N (nodus) – 0,3496.

M (metastasis) – 0,21074.

На основании полученных коэффициентов корреляции составлена следующая формула совокупной оценки наиболее значимых факторов 1 ( $M_1$ ):

$$M_1 = T + N + M + S \quad (1)$$

Далее методом простой линейной регрессии рассчитаны коэффициенты k и b и выведена формула продолжительности жизни (y) на основании значения метрики:

$$y = -115 \cdot x + 1388 \quad (2),$$

где k = -115; b = 1388.

Исходя из проведенного исследования можно сделать вывод, какие факторы являются ключевыми для продолжительности жизни. Полученные модель и приемы могут использоваться при первичной обработке данных об онкологических заболеваниях и для построения экспресс-прогноза течения заболевания. Для обеспечения возможности анализа больших объемов данных требуется использование вычислительных средств, из чего можно сделать вывод о необходимости разработки соответствующего программного средства. Необходимость о такой разработке заявлена со стороны Минского городского клинического онкологического диспансера. В рамках дипломного проекта планируется разработать соответствующее программное средство и продолжить работу над исследованиями.

**Список использованных источников:**

1. Данные ВОЗ по заболеваемости раком за 2014 год. Беларусь | Официальный сайт Всемирной организации здравоохранения [Электронный ресурс]. Режим доступа: [https://www.who.int/cancer/country-profiles/blr\\_ru.pdf?ua=1](https://www.who.int/cancer/country-profiles/blr_ru.pdf?ua=1) – Дата доступа: 10.04.2020.

## **ВРМ СИСТЕМЫ КАК ФАКТОР ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ ПРЕДПРИЯТИЯ**

*Манцевич Андрей Валерьевич*

*Белорусский Государственный Университет Информатики и Радиозлектроники,  
кафедра менеджмента, г. Минск, Республика Беларусь*

*Домакур О.В. – канд. эконом. Наук*

Врм системы, необходимость применения врм систем в структуре предприятия, практическая эффективность на примере предприятия.

В текущее время эффективность любого предприятия зависит не только от его производственной мощности, квалификации сотрудников, обеспеченности в финансировании и инвестициях, но также и в правильном и рассчитанном менеджменте, контроле и управлении всеми его структурными подразделениями.

Любому предприятию необходимо развиваться в условиях современной экономики, когда конкуренция постоянно растет, а количество сторонних предложений растет. Однако необдуманно подходить к данному процессу невозможно, необходимо четкое планирование и моделирование всех

возможных путей модификации предприятия. Специально для этого и создаются bpm системы.

BPM системы – это класс программных продуктов, которые помогают управлять бизнес процессами организации. Данное программное обеспечение позволяет смоделировать каждый отдельный процесс в компании, представить его в графической схеме. Также позволит управление, анализ и дальнейшее совершенствование бизнес процессов путем моделирования возможных изменений в процессе в зависимости от основных критериев.

BPM системы предназначены для повышением эффективности работы организации, путем реорганизации процессов и улучшения системы управления. Этот класс программного обеспечения создавался с таким расчетом, чтобы можно было быстро определить слабые места процессов и подразделения компании, которые снижают общую эффективность работы, проанализировать их, и выдать расчетные модели по возможности улучшения ситуации.

Среди всех специализированных информационных систем, позволяющих анализировать, моделировать, управлять и отображать бизнес-процессы предприятия, BPM системы находятся на стыке трех сфер корпоративного программного обеспечения (рис. 1).



Рисунок 1 – Сочетание классов ПО в BPM системе

После внедрения bpm систем в структуру управления предприятием, отмечалось увеличение гибкости в области принятия решений, уменьшение нагрузки на отделы анализа и планирования.

При проведении математического расчета при моделировании коэффициент ускорения принятия решений на основе работы системы составляет 1.63х от стандартного. Хотя данные были собраны автором только на одном предприятии, следует предположить, что система может эффективно показать себя и на других предприятиях.

#### **Вывод**

Необходимо привлечение bpm систем с структуру управления предприятия и увеличение общего уровня заинтересованности специалистов в работе с данными системами. Этого можно добиться субсидированием в сферу образования на уровне университетов, путем предоставления учащимся возможности изучать материалы и пользоваться существующими платформами для создания bpm систем. А также использованием bpm систем на различных предприятиях, в качестве примера эффективности данных систем.

#### **Список использованных источников:**

1. Pega Platform Foundation Exercise Guide / Pegasystems Inc. One Rogers Street, 2010.
2. System Architect Essentials Exercise Guide/ One Rogers Street Cambridge, MA 02142-1209, 2009.

## **ПРОГРАММНОЕ СРЕДСТВО ФОРМИРОВАНИЯ ОБУЧАЮЩЕЙ ТАКТИКИ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННОЙ СЕТИ**

*Кузьмичкин Н.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Куликов С.С. – к.т.н., доцент*

В наше время игры имеют большое значение, создается большое количество кооперативных игр, игроки соревнуются друг с другом. Киберспорт набирает большую популярность. В любом виде спорта требуется тренер, а в киберспорте их дефицит. В играх обучение происходит на базовом уровне. И тут возникает проблема, где же взять того самого тренера? В решении этого вопроса нам поможет искусственный интеллект.

Основной задачей программного средства является анализирование и нахождение стратегий и навыков, способных повысить игровой уровень для победы, с последующим обучением игрока

найденным решениям.

Использование нейронных сетей в анализе и нахождении обучающей тактики может идти по двум стратегиям:

- 1) сбор и обработка данных игровых моментов
- 2) создание игрока с искусственным интеллектом.

Сбор и обработка данных игровых моментов. Стратегия включает использование нейронных сетей для “интеллектуальной” обработки статистической информации, получаемой от пользователя с дальнейшим включением накопленной информации в шаблоны. В идеальном варианте анализируется весь игровой процесс. Но так как анализ игрового процесса в большинстве игр задействует большое количество ресурсов, наиболее рационально использовать только ключевые моменты игры. Из моментов ведется поиск шаблонов желательных и нежелательных действий (действия ведущие к выигрышу и проигрышу соответственно).

Минус данной стратегии в том, что все игроки разные, у каждого есть свои собственные предпочтения в ритме и стиле игрового процесса и количестве возможностей, которые дарует геймплей. Эта разница может быть крайне большой.

Создание игрока с искусственным интеллектом. Данная методика дает игроку с ИИ список возможных к использованию функций. При получении игроком с ИИ входной информации, производится решение о выборе или включении той или иной функции. Каждый нейрон суммирует значения всех нейронов из предыдущего слоя, умноженные на вес соединения. Затем последний слой производит вывод. Можно сказать, что наш игрок с ИИ, как новорожденный, который учиться играть, но с большей скоростью.

Одним из важных аспектов стратегии игрока с искусственным интеллектом является менеджер поколений. На одно событие создается несколько копий игроков с ИИ со случайной мутацией между нейронами в нейронной сети. Мутация весов синапсов влияет на выбор функции и время ее выполнения. После окончания события менеджер поколения будет использовать нейронную сеть наиболее приспособленного игрока с ИИ.

Минусы стратегии в большой ресурсозатратности при большом количестве функций. А также в некоторых играх большое количество поколений.

Следующей задачей программного обеспечения является обучение игрока. Обучить игрока можно следующими методами:

- 1) Использование оповещений о необходимости в тех или иных действиях.
- 2) Использование полупрозрачной копии в небольшом временном опережении.
- 3) Использование повторяющейся тренировки решающих событий.

Если с оповещением о необходимости в тех или иных действиях все понятно, то при использовании полупрозрачной копии в небольшом временном опережении создается полупрозрачный игрок с ИИ, который выполняет действия считающиеся желательными в той или иной ситуации, но сам он никак не влияет на процесс игры.

При использовании повторяющейся тренировки решающих событий с помощью нейронной сети находятся ключевые моменты. И игрок должен пройти ключевой момент пока не достигнет желательного результата или результата близкого к шаблону. При невыполнении данной миссии событие начинается заново. Напоминает “День Сурка”.

При правильном формировании обучающей тактики и обучении игроков могут получить выгоду как игроки, поднимающие свои навыки, так и компании, удерживающие игроков возможностью развиваться. Также тренер с ИИ способствует развитию киберспорта, снижению затрат на формирование обучающихся уровней.

**Список использованных источников:**

1. <https://habr.com/ru/post/312450/>
2. <https://cyberleninka.ru/article/n/vozmozhnoe-primeneniye-neyronnyh-setey-v-igrah-v-buduschem/viewer>
3. <http://stevenmiller888.github.io/mind-how-to-build-a-neural-network/>

**СЕКЦИЯ «ВЫСШАЯ МАТЕМАТИКА»**

## МАТЕМАТИКА И МАТЕМАТИКИ В АНЕКДОТАХ И ШУТКАХ ЧАСТЬ 1, 2

Акулович Г.Э., Кисин К.А., Юрчик В.А., Якушев И.Е.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Борисенко О.Ф. – канд. физ.-мат. Наук, доцент

В докладе представлен анализ математического юмора.

*Исследование ввелось нами в трех направлениях:*

- Юмор, который чему-то учит, а также интересные факты
- Анекдоты и забавные истории из преподавания и изучения математики
- Реальные истории преподавателей и великих математиков

Юмор положительно влияет на все аспекты человеческой жизни; преподавание и изучение математики не стали исключением. Несмотря на то, что математический юмор местами специфичен и сложен для обычного слушателя, он часто может завлечь и заинтересовать. Также он показывает математиков и слуг науки с другой, интересной и необычной стороны.

Без юмора человеческая жизнь была бы гораздо сложнее. Также ко всему пласту юмора, а именно к шуткам, забавным историям, юморескам, и просто к бородатым анекдотам от которых становится тепло на душе, не принято относиться со всей серьезностью, ибо появится огромный перечень людей, профессий и даже целых наций, для которых это может звучать оскорбительно, так как часто шутки строятся на абсурде или человеческой глупости.

Юмор в математике стал неким исключением, ибо хороший анекдот или факт может не только рассмешить, но и чему-то научить, даже заинтересовать обывателя близко не знакомого с константами и уравнениями. Например, существует шутка, что  $e$  это  $\pi$  наоборот. Количество цифр после запятой как у  $e$ , так и у  $\pi$  является бесконечным, в этом ряде цифр есть все их сочетания, например, дата вашего рождения или номер вашего телефона; Не исключено, что в конце этой бесконечности (если у бесконечности может быть конец) вполне может быть число  $e$  записанное наоборот.

Или забавный факт с тем же числом  $\pi$ . Он звучит так: 45 сантиметровая пицца, больше чем две тридцатисантиметровые.

Что касается анекдотов про лекции или практические занятия, то это как правило истории про невнимательность студентов или про их незнание, или же острые шутки, оговорки преподавателей.

Бытуют два мнения: Первое. Что занятия должны быть серьезными, чтобы студенты не отвлекались от материала. И второе, которое разделяем мы. Анекдоты и истории разряжают обстановку на лекциях и позволяют хоть на несколько секунд передохнуть и с новыми силами взяться за материал. А некоторые из них, как своего рода мнемонические правила помогают запомнить какие-то формулы, усвоить материал, который в погоне за знаниями может преподаваться в сухой, на первый взгляд, неинтересной форме.

### Список использованных источников:

1. Федин. Математики тоже шутят. Изд-во Книжный дом «ЛИБРОКОМ» 2009. – 216 с.
2. Персональный сайт Барвенова С.А. <http://bars-minsk.narod.ru/>
3. Прохорович М.А. Курьёзы и юмор с физико-математическим уклоном. ООО «Фотон век», Пущино, 2015, 216 с.

## ОПТИМИЗАЦИЯ ПРОЦЕССА ДЕЛЕНИЯ ИГРОКОВ НА ТРЕНИРОВКАХ ПО БАСКЕТБОЛУ

Беззубенко Е.Д., Маскалик Д.Д.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Фомичева Л.А. – канд. физ.-мат. наук

В работе описана программа, выполняющая деление на команды игроков на любительских тренировках по баскетболу. Программа реализована в Excel в среде Visual Basic for Applications. Может применяться для деления команд на тренировках по другим игровым видам спорта.

В любительских командах по баскетболу деление игроков на тренировках осуществляется

различными способами. Например, распространенным является способ, при котором игроки, назначенные капитанами, по очереди набирают себе игроков. При этом, как правило, каждый раз формируются команды с похожими составами. Если стоит цель разнообразить составы, осуществить ротацию игроков, а также исключить личностный фактор при формировании команд, то одним из способов является использование компьютерной программы, позволяющей выполнить беспристрастное деление игроков на команды.

Критериев, по которым может быть составлен алгоритм работы программы может быть множество. В профессиональных командах по игровым видам спорта используются программы со сложным алгоритмом, в которых учитывается на какой позиции играет спортсмен, какой процент попаданий, уровень дриблинга, подборки, если речь идет о баскетболе и гандболе, и прочее. В них также учитывается, как сочетаются на разных позициях те или иные игроки. С помощью подобных программ формируются оптимальные составы команд для соревнований.

На тренировках любительских команд ставится цель получить приблизительно одинаковые по уровню игры команды. В интернете можно найти различные приложения, в которых можно указать количество игроков, расставить рейтинги, указать количество игроков в команде и получить списки игроков для тренировки. Но в каждой из них есть свои нюансы. Например, если на тренировке 13 человек, а в программе указать, что в команде должно быть пять игроков, то программа делит на три команды, но при этом две команды будут по 5 человек, а в одной будет три игрока, при этом рейтинг уже не учитывается. Есть программы, которые произвольным образом делят игроков на команды без привязки к рейтингу. После тестирования различных приложений, был сделан вывод, что ни одна из программ, имеющихся в свободном доступе, не соответствует нужным нам критериям.

Нами была написана программа в Excel (Visual Basic for Applications). За основу работы программы были взяты две характеристики: рейтинг игрока и его рост. Рейтинг игрока определяется по десятибалльной бальной шкале. В первую очередь учитывается точность бросков, уровень дриблинга, скорость, выносливость, общие игровые навыки. С помощью анонимного опроса среди постоянных участников тренировки был составлен рейтинг игроков. В таблицу в Excel были внесены данные игроков с указанием их рейтинга и роста.

Программа случайным образом подбирает комплект игроков таким образом, чтобы разница суммарного рейтинга команд не превышала заданного отклонения, значение которого может задать пользователь. Вторым критерием является рост игроков. Пользователь может задать максимально допустимую разницу в среднем росте команд. Эту опцию можно отключить. Данная возможность была добавлена, поскольку при неодинаковом количестве игроков в команде данный критерий не позволяет добиться оптимальных составов.

Программа была протестирована на тренировках любительской женской команды «Космос», которая проводит тренировки в СОЦ БГУИР. Применение программы позволило улучшить психологический климат в коллективе, а также сэкономило время, которое раньше тратилось на деление команд и споры о равнозначности составов команд. Поскольку в процессе тестирования программы появлялись разного рода нюансы, например, несовместимость отдельных игроков, были добавлены дополнительные опции, позволяющие сделать так, чтобы конкретные игроки не попадали в одну команду или наоборот, конкретно на этой тренировке играли в одной команде. В целом игроки дают положительные отзывы о программе и признают, что программа выдает оптимальные составы для игры.

Программа может применяться на занятиях по физической культуре в школах и университетах.

**Список использованных источников:**

1. Новиков, Д.А. Математические модели формирования и функционирования команд/Д.А. Новиков. – М.: Издательство физико-математической литературы, 2008. – 184 с.

## **ПРИМЕНЕНИЕ МАТЕМАТИЧЕСКИХ МЕТОДОВ СТАТИСТИКИ В МЕДИЦИНЕ**

*Верига М.Е.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Фомичева Л.А. – канд. физ.-мат. наук, Новоротская Н.Л. – ассистент*

В работе с помощью методики М. Люшера исследовано психоэмоциональное состояние студентов первого курса Факультета компьютерного проектирования БГУИР. Сделаны выводы о степени эмоционального дискомфорта данной выборки.

В 1949 году швейцарский психотерапевт Макс Люшер опубликовал тест, основанный на установленной им после множества экспериментов зависимости между предпочтением

определенных цветов и текущим психологическим состоянием человека. С точки зрения математики тест Люшера интересен тем, что позволяет вычислить количественные показатели тревожности, эмоциональной стабильности и стрессоустойчивости. Среди них – суммарное отклонение от аутогенной нормы (СО), вегетативный коэффициент (ВК) и некоторые другие показатели.

Для расчета СО сравнивают порядок цветов, указанный испытуемым, с их «идеальным» расположением – 34251607 - где цифры соответствуют цветам следующим образом: 3 – красный, 4 – желтый, 2 – зеленый, 5 – фиолетовый, 1 – синий, 6 – коричневый, 0 – серый, 7 – черный. Затем находят разницу между занимаемым местом и нормативным положением цвета и суммируют модули этих разностей. Эта сумма и есть СО. Ее значение может быть только четным и лежит в пределах от 0 до 32.

Вегетативный коэффициент устанавливает, влияние какой вегетативной нервной системы – симпатической или парасимпатической – преобладает. Преобладание симпатической ВНС ( $ВК > 1,2$ ) отражает перевозбуждение, склонность к лихорадочности или зажатости, стремление к затрате энергии, тогда как преобладание парасимпатической ВНС ( $ВК < 1,2$ ) говорит об усталости, склонности к сохранению, накоплению сил и означает установку на минимизацию усилий, неготовность к активной деятельности. Расчет:  $ВК = (18 - К - Ж) / (18 - С)$ , где: К, Ж, С, 3 — места красного, желтого, синего и зеленого цветов соответственно. Для расчета ВК в данной работе используются методики диагностики и измерения психических состояний личности А.О.Прохорова.

Нами было проведено исследование среди студентов первого курса ФКП БГУИР. Участие в нем приняли 182 человека в возрасте от 17 до 19 лет.

Результаты исследования по суммарному отклонению от аутогенной нормы приведены на рисунке 1:

### Суммарное отклонение

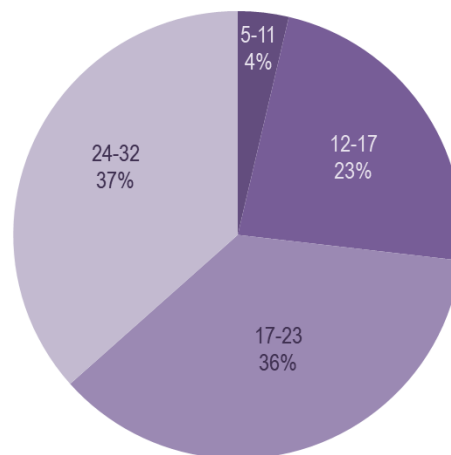


Рисунок 1 – Суммарное отклонение от аутогенной нормы

Поясним полученные результаты.

4% опрошенных набрали от 5 до 11 баллов, что говорит о хорошей устойчивости, установке на активность, способности свободно управлять свои вниманием.

23% испытуемых оказались в диапазоне от 12 до 17. Для этого промежутка характерны способность к переходу от одного вида деятельности к другому без существенных затруднений и преодолению усталости усилием воли, однако лишь на некоторое время, по истечению которого работоспособность снижается на долгий период.

Почти поровну (36 и 37% соответственно) опрошенных оказались в промежутках 17-23 и 24-32. У опрошенных, находящихся в первом из этих промежутков, отмечают потребность в постоянном самоконтроле, помогающем им против воли выполнять необходимые действия, и отсутствие удовлетворения от рабочего процесса и результатов, которые в совокупности повышают психическое напряжение и вызывают переутомление. Для состояния тестируемых из второго промежутка характерны высокая утомляемость, низкая концентрация внимания, колебание настроения от восторженного до подавленного и обратно, раздражительность, беспочвенная тревога, предчувствие неприятностей, отсутствие желания делать что-либо.

Среди всех участников опроса ни один человек не попал в промежуток от 0 до 5 баллов, состояние в котором описывается как деятельное и эффективное, с преобладанием оптимистичного настроения, а также способностью к длительной интенсивной работе и мобилизации в экстренной ситуации.



Результаты расчёта вегетативного коэффициента отражены на рис. 2:

### Вегетативный коэффициент

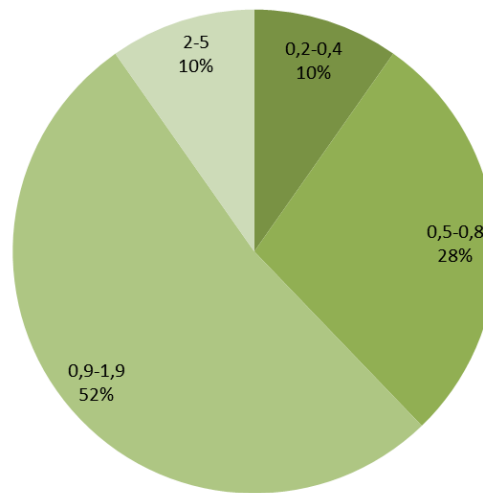


Рисунок 2 – Вегетативный коэффициент

Результаты, приведенные на рисунке 2, делятся на четыре группы.

В первую (0,2 – 0,4) попали 10%. Для этой группы характерны хроническое переутомление, пассивность, неготовность к действиям в экстренной ситуации, склонность к бездействию.

Во второй группе (0,5 – 0,8) – 28%, чье состояние описывается как склонность к оптимальному расходованию сил, способность к эффективной деятельности в привычных условиях, но затруднения с принятием решений в экстренной ситуации.

Более половины – 52% - оказались в третьей группе (0,9 – 1,9), что не может не радовать, так как это состояние баланса двух ВНС, в котором человек мобилизован, склонен к активным действиям, для него свойственна высокая скорость принятия решений, его действия целесообразны, эффективны.

Наконец, 10% опрошенных попали в четвертую группу (2,0 - 5). Такое количество баллов говорит о перевозбуждении, суетливости, нетерпеливости, импульсивности, лихорадочности реакций и низкой эффективности действий в стрессовых ситуациях.

Методику Люшера предпочитают другим, так как она может за короткое время (меньше 5 минут) дать глубокую и обширную характеристику склонности испытуемого к какой-либо деятельности, его настроения и наиболее устойчивых черт его личности, притом тестируемый не сможет повлиять на результат сознательно.

Таким образом, использование методов математической статистики позволило нам изучить и проанализировать полученные данные. Результаты исследования могут стать основой для дальнейшей работы специалистов в области психологии и психотерапии.

#### Список использованных источников:

1. Тест Люшера [Электронный ресурс]. – 2020. – Режим доступа: <https://4brain.ru/blog/luscher-test/>
2. Математика Люшера [Электронный ресурс]. – 2020. – Режим доступа: <https://cutt.ly/0yq90vO>
3. Цветовой тест Люшера. Полный вариант методики. [Электронный ресурс]. – 2020. – Режим доступа: <https://psycabi.net/testy/553-tsvetovoj-test-lyushera-polnyi-variant-metodiki>
4. Цветовой тест Люшера [Электронный ресурс]. – 2020. – Режим доступа: <https://cutt.ly/Yyq3rqD>

## ЦЕПОЧКИ ОКРУЖНОСТЕЙ, ВПИСАННЫХ В ПАРАБОЛУ

Зезюлькин П.А., Ващилин Е.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Баркова Е.А. – канд. физ.-мат. наук, доцент

Япония находилась в многовековой изоляции. Однако это не помешало развитию японской науки, в частности, математики. Японцы полагали, что искусство геометрии угодно Богу, и ею увлекались представители всех сословий. Много работ было посвящено кривым второго порядка, а

особое внимание уделялось эстетике открытий. Поэтому для данной работы мы сформулировали следующую задачу: возможно ли вписать цепочки окружностей в параболу, и будут ли полученные закономерности лаконичными?

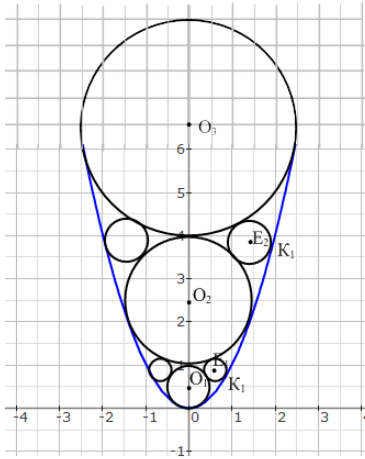


Рисунок 1

В данной работе нами рассматриваются две цепочки окружностей, вписанных в параболу: первая цепочка – это окружности  $\omega_n$ , центры которых находятся на оси  $Oy$  (на рисунке 1 это точки  $O_1, O_2, \dots, O_n$ ) и которые касаются друг друга и параболы. Вторая цепочка – это окружности  $\psi_n$ , вписанные в криволинейные треугольники, образованные окружностями из первой цепочки и ветвями параболы. Их центры – это точки  $E_1, E_2, \dots, E_n$  на рисунке 1, а  $K_1, K_2, \dots, K_n$  – точки касания этих окружностей с параболой.

Найдём радиусы и координаты центров окружностей из первой цепочки. Для этого составим системы из уравнений окружностей и параболы для окружностей  $O_1$  и  $O_2$ :

$$\begin{cases} y = x^2, \\ x^2 + (y - R_1)^2 = R_1^2 \end{cases} \quad (1), \quad \begin{cases} y = x^2, \\ x^2 + (y - (2R_1 + R_2))^2 = R_2^2 \end{cases} \quad (2)$$

С помощью арифметической прогрессии найдём координаты центра  $n$ -ой окружности:  $O_n(O; n^2 - n + \frac{1}{2})$ . На основании полученных решений для систем (1) и (2) ( $R_1 = 0.5; R_2 = 1.5$ ) возникает гипотеза о том, что радиус  $n$ -ой окружности можно найти по формуле  $R_n = n - \frac{1}{2}$ . Данную формулу нетрудно доказать методом математической индукции.

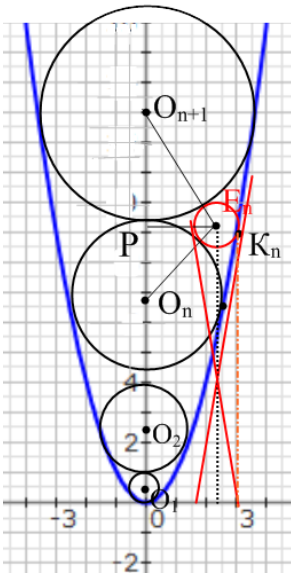


Рисунок 2

Рассмотрим рисунок 2. Пусть  $E_n(x_{1n}; y_{1n})$  — центр окружности  $\psi_n$ , т.е.  $n$ -ой окружности, вписанной в криволинейный треугольник. Эта окружность касается окружностей  $\omega_n$  и  $\omega_{n+1}$ , т.е.  $n$ -ой и  $(n + 1)$ -ой окружностей с центрами на оси  $Oy$ . Пусть  $r_n$  – радиус этой окружности.

$K_n(x_{2n}; y_{2n})$  — точка касания  $n$ -ой окружности и параболы;  $y_{2n} = x_{2n}^2$ .

$O_{n+1}(O; y_{n+1})$  — центр  $(n+1)$ -ой окружности с центром на оси  $Oy$ .

$O_n(O; y_n)$  — центр  $n$ -ой окружности.

Проведём  $E_nP \perp Oy$ . Из треугольников  $O_{n+1}E_nP$  и  $O_nE_nP$ :

$$\begin{cases} O_{n+1}P^2 + PE_n^2 = O_{n+1}E_n^2 \\ O_nP^2 + PE_n^2 = O_nE_n^2 \end{cases}, \text{ или } \begin{cases} (y_{n+1} - y_{1n})^2 + x_{1n}^2 = (R_{n+1} + r_n)^2 \\ (y_n - y_{1n})^2 + x_{1n}^2 = (R_n + r_n)^2 \end{cases} \quad (3)$$

Из системы (3) получим:

$$r_n = 2n^3 - 2ny_{1n} \text{ или } y_{1n} = \frac{2n^3 - r_n}{2n}.$$

С помощью пары касательных к окружности  $\psi_n$ , симметричных относительно прямой  $x = x_{1n}$ , проходящей через центр данной окружности, получим выражение для абсциссы центра  $\psi_n$  (формула 5) и её радиуса (формула 6).

$$x_{1n}^2 = 9r_n^2 - \frac{9}{64} \quad (5), \quad r_n = \frac{n}{4} \quad (6).$$

Таблица 1 – Результаты исследования

Цепочка окружностей	Координаты центра	Радиус
$O_n$	$O_n(O; n^2 - n + \frac{1}{2})$	$R_n = n - \frac{1}{2}$
$\psi_n$	$E_n(\frac{3}{4} \sqrt{n^2 - \frac{1}{2}; n^2 - \frac{1}{8}})$	$r_n = \frac{n}{4}$

Также в ходе работы мы нашли координаты точки  $K_n$  касания окружности  $\psi_n$  и параболы:

$$K_n \left( \sqrt{n^2 - \frac{1}{4}}; n^2 - \frac{1}{4} \right) \quad (7)$$

В силу симметрии данную задачу можно рассматривать и в пространстве, а именно вписать цепочки сфер в параболоид (рисунки 3, 4).

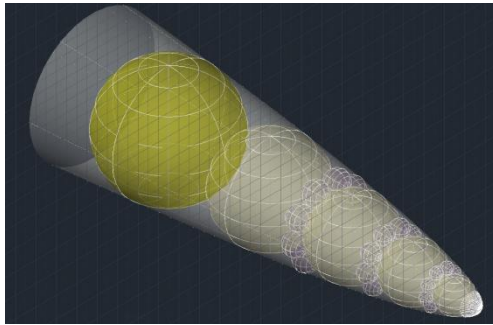


Рисунок 3

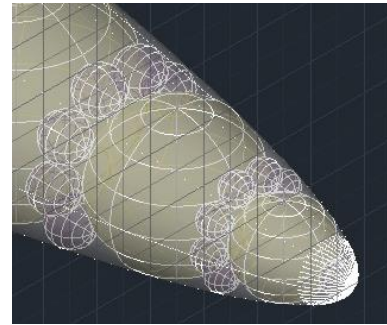


Рисунок 4

Рассматривая полученные цепочки, необходимо обратить внимание на возможность вписания окружности в криволинейный треугольник, образованный тремя окружностями. Для этого используем теорему Декарта, согласно которой радиусы четырёх взаимно касающихся окружностей удовлетворяют некоторому квадратному уравнению, а именно:

$$(k_1 + k_2 + k_3 + k_4)^2 = 2(k_1^2 + k_2^2 + k_3^2 + k_4^2) \quad (8),$$

где  $k_i = \frac{1}{r_i}$  - кривизна окружности.

Чтобы найти координаты центра такой окружности, можно рассмотреть т. Декарта в комплексном виде, представив координаты  $(x, y)$  центра в виде комплексного числа  $z = x + iy$ . Тогда координаты центра  $z_4$  можно найти по формуле:

$$z_4 = \frac{z_1 k_1 + z_2 k_2 + z_3 k_3 \pm 2\sqrt{k_1 k_2 z_1 z_2 + k_2 k_3 z_2 z_3 + k_1 k_3 z_1 z_3}}{k_4} \quad (9)$$

Таким образом у нас появилась возможность построить третью цепочку окружностей. Полученные окружности выделены красным цветом на рисунке 5.

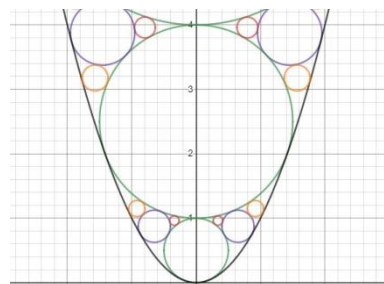


Рисунок 5

**Список использованных источников:**

1. Гусак А.А. Справочник по высшей математике / А.А. Гусак, Г.А. Гусак, Е.А. Бричкова. – Минск: ТетраСистемс, 1999. – 640 с. ISBN: 985-6317-51-7.
2. Журнал "Квант", №4 – 2013
3. Beyond the Descartes circle theorem / Colin L.Mallows [et al.] // Amer. Math. Monthly 109 (2002), 338--361.
4. Japanese Temple Geometry / Rothman T. // Scientific American, 278, #5 1998, p.85-91.

# ОПТИМИЗАЦИЯ ПОТРЕБИТЕЛЬСКОГО ВЫБОРА СОВРЕМЕННЫХ СТУДЕНТОВ НА ОСНОВЕ МНОГОПРОДУКТОВОЙ ФУНКЦИИ ПОЛЕЗНОСТИ ТИПА КОББА-ДУГЛАСА

Клюй В.И., Корбут К.И.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Ефремов А.А. – канд. экон. наук,  
доцент кафедры ВМ БГУИР

В работе приведено исследование максимизации общей полезности студента при заданном бюджетном ограничении для случая функции полезности типа Кобба-Дугласа. Работа выполнена на реальных данных, полученных посредством социологического опроса.

Концепция бюджетных ограничений в области экономики вращается вокруг идеи, что данный потребитель ограничен в потреблении относительно количества капитала, которым он обладает. В результате потребители анализируют оптимальный способ использования своей покупательной способности для максимизации своей полезности. Это достигается за счет использования бюджетных ограничений, которые представляют собой вероятные комбинации продуктов и / или услуг, которые покупатель может приобрести за свой капитал.

Тем не менее, многие экономические расчеты ограничены только двумя продуктами, и полученные цифры могут быть далеки от реальности.

Цель исследования - собрать студентов разных типов и выяснить, как они могут наиболее эффективно использовать свой бюджет. Полезность субъективна, поскольку определяется индивидуальными предпочтениями того или иного индивида. В связи с чем коэффициенты полезности, использованные в следующих расчетах, были получены экспертным путем на основе опроса 96 студентов от 17 до 22 лет. Основные вопросы анкеты:

1. К какому типу студента вы себя относите?
2. Используя 5-балльную систему оцените полезность 6-ти следующих продуктов: кино, обед, тренажерный зал, фаст-фуд, продукты гигиены, одежда.

Таблица 1 – Данные для студентов типа «Домосед» и «Активный»

Товары	Цена, BYN	Кoeff. полезности для студентов типа «Домосед»	Кoeff. полезности для студентов типа «Активный»
1. Кино	6,00	2,88	3,93
2. Обед	4,00	3,94	4,38
3. Тренажерный зал	8,00	3,00	3,5
4. Фаст-фуд	2,00	4,00	3,88
5. Продукты гигиены	2,33	1,39	1,5
6. Одежда	6,60	1,12	1,46

Существуют различные способы задания функции полезности:

1. линейная функция для совершенных субститутов:

$$u(x, y) = \alpha x + \beta y \quad (1),$$

2. функция Леонтьева для совершенных complements:

$$u(x, y) = \min \{ \alpha x, \beta y \} \quad (2),$$

3. функция Кобба-Дугласа:

$$u(x, y) = x^\alpha y^\beta; \quad (3),$$

В вычислениях используем функцию типа Кобба-Дугласа.

При данном анализе получаются "хорошие" предпочтения, т.е. строго монотонные и строго выпуклые. Также это является одним из самых простых и удобных для вычислений способов задать предпочтения потребителей.

Так, общая полезность для студентов типа «домосед» может быть рассчитана на основе данных таблицы 1:

$$U = x_1^{2,88} x_2^{3,94} x_3^3 x_4^4 x_5^{1,39} x_6^{1,12} \quad (4),$$

Для студентов типа «активный» формула следующая:

$$U = x_1^{3,93} x_2^{4,38} x_3^{3,5} x_4^{3,88} x_5^{1,5} x_6^{1,46}. \quad (4)$$

При этом важно заметить, что данный потребитель не может отказаться от одного блага (в пользу другого), т.к. функция полезности в таком случае обнулится.

В расчетах также учитывается бюджетное ограничение, которое рассчитывается по формуле:

$$x_1 p_1 + x_2 p_2 + x_3 p_3 + x_4 p_4 + x_5 p_5 + x_6 p_6 + x_7 p_7 \leq I. \quad (5)$$

Компьютерная реализация экономико-математической модели, представленной выше, с использованием инструмента «Поиск решения» в MS Excel, дала следующие результаты:

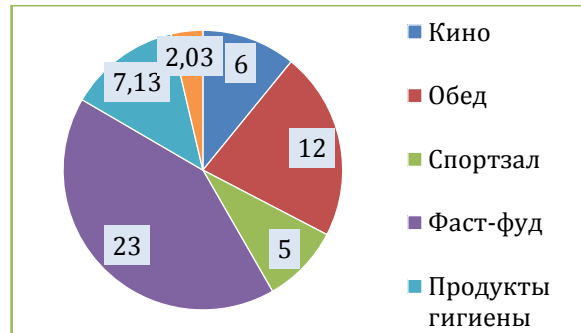


Рисунок 1: Результаты использования инструмента «Поиск решения» в MS Excel для студентов типа «Домосед»

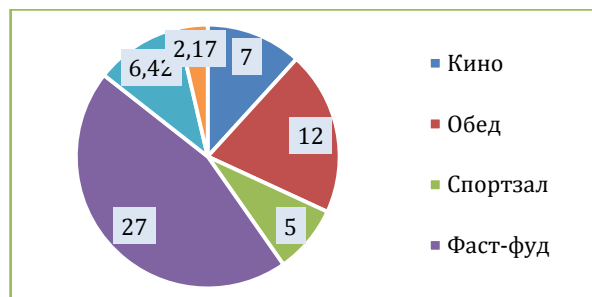


Рисунок 2: Результаты использования инструмента «Поиск решения» в MS Excel для студентов типа «Активный»

1. Студенты-домоседы могут максимизировать свою полезность, используя следующую комбинацию продуктов и услуг: кинотеатр - 6, обед - 12, тренажерный зал - 5, закуски - 23, средства по уходу - 7,13, одежда - 2,03.

2. Активные студенты могут максимизировать свою полезность, используя следующую комбинацию продуктов и услуг: кинотеатр - 7, обед - 12, тренажерный зал - 5, закуски - 27, средства по уходу - 6,42, одежда - 2,17.

3. При сравнении показателей двух типов студентов можно сделать вывод о том, что при различных предпочтениях оба студента могут позволить себе одинаковое количество обедов и посещений тренажерного зала, в то время как студентам типа «Активный» для максимизации полезности нужно больше посещений кино, а также покупки большего количества фаст-фуда и одежды.

Функция полезности является очень удобным вспомогательным средством, которое открывает возможность использования теории оптимизации при решении задачи потребителя. Без использования функции полезности решение такой задачи с математической точки зрения может быть затруднительным.

В заключение следует отметить, что экономико-математическая модель, представленная в этой статье, также может быть использована для оценки целесообразности расходов разных типов людей. Такие задачи в настоящее время достаточно актуальны и представляют практический интерес.

**Список использованных источников:**

1. Библиотека Гумер [Электронный ресурс] – Режим доступа: <https://www.gumer.info>.
2. Wikipedia, Consumer choice [Электронный ресурс] – Режим доступа: [https://en.wikipedia.org/wiki/Consumer\\_choice](https://en.wikipedia.org/wiki/Consumer_choice).
3. Studme [Электронный ресурс] – Режим доступа: [https://studme.org/1451060519625/ekonomika/ravnovesie\\_potrebitya](https://studme.org/1451060519625/ekonomika/ravnovesie_potrebitya).

# НАТУРАЛЬНЫЕ ТОЧКИ ПОД КРИВОЙ

Шедов В.С.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Примичева З.Н. – к.ф.-м.н., доцент

В данной работе исследовано количество натуральных точек под различными видами кривых. Рассмотрены кривые:  $y = ax$ ,  $y = ax^2$ ,  $y = a\sqrt{x}$ ,  $y = ax^{\frac{1}{3}}$ ,  $y = ax^3$ ,  $y = ax^{\frac{1}{m}}$ ,  $y = ax^m$ ,  $y = \log_a x$ ,  $y = \alpha \sin x$ ,  $y = \alpha \arcsin x$ , обобщение в пространстве для плоскости  $z = ax + by$ , где  $\alpha$  и  $\beta$  – положительные действительные числа. Для этих кривых найдены пределы и формулы описывающие свойства количества точек под ними.

I. Рассмотрено множество натуральных точек под прямой  $y = ax$ , которое обозначено  $f_\alpha(n)$ .

1.1. а) Для нахождения  $f_{\sqrt{2}}(6)$  на координатной плоскости  $Oxy$  рассмотрена область, ограниченная прямыми  $y = \sqrt{2}x$  и  $x = 6$ . Количество точек с натуральными координатами в рассматриваемой области  $N = 27$ . Для  $f_{\sqrt{3}}(4)$ ,  $N = 15$ .

б) Для нахождения  $f_1(n)$  на координатной плоскости  $Oxy$  рассмотрена область, ограниченная прямыми  $y = x$  и  $x = n$ .  $f_1(n) = \frac{n(n+1)}{2}$ . Для  $\alpha > 0$ :

$$N = \sum_{x=1}^n [\alpha x]. \quad (1)$$

1.2. Доказано, что ни при каких  $\alpha$  и  $\beta$  равенство  $f_\alpha(n) = f_\beta(n)$  не выполняется.

1.3. Вычислен  $f_\alpha(n)$  для различных  $\alpha$ . При целом  $\alpha$  формула  $N = \sum_{x=1}^n [\alpha x]$  приобретает вид

$$N = \sum_{x=1}^n \alpha x = \frac{(\alpha+an)n}{2}. \quad (2)$$

При все остальных  $\alpha$  используем формулу  $N = \sum_{x=1}^n [\alpha x]$ .

1.4. Вычислен предел:  $\lim_{n \rightarrow \infty} \frac{f_\alpha(n)}{n^2} = \frac{\alpha}{2}$ .

1.5. Доказано, что:

а)  $f_\alpha(n) = \frac{1}{2}[\alpha]n(n+1) + f_{\{\alpha\}}(n)$ , где  $[a]$  - целая часть,  $\{\alpha\}$  - дробная часть числа  $\alpha$ .

б)  $f_\alpha(n) + f_{\frac{1}{\alpha}}([n\alpha]) - \left[\frac{n}{\alpha}\right] = n[n\alpha]$ . Построен аналог этой формулы для произвольного положительного  $\alpha \in \mathbb{R}$ .

1.6. а) Проверено выполнение равенства  $f_{\alpha+\beta}(n) = f_\alpha(n) + f_\beta(n)$  для целых  $\alpha$  и  $\beta$ . Найдено всё множество пар  $(\alpha, \beta)$ , для которых это равенство будет выполняться для любого натурального  $n$ . Найдена связь между  $f_{\alpha+\beta}(n)$  и  $f_\alpha(n), f_\beta(n)$ . Доказаны утверждения  $f_{\alpha+\beta}(n) = f_\alpha(n) + f_\beta(n)$  при  $\alpha \in \mathbb{Z}, \beta \in \mathbb{Z}, \alpha\beta \geq 0$  и  $f_{\alpha+\beta}(n) < f_\alpha(n) + f_\beta(n)$  при  $\alpha \in \mathbb{Z}, \beta \in \mathbb{Z}, \alpha\beta < 0$ .

б) Найдена связь (в виде нетривиальных тождеств или неравенств) между  $f_\alpha(n+m)$  и  $f_\alpha(n), f_\alpha(m)$ .

II. Множества натуральных точек под параболой  $y = ax^2$ ,  $y = a\sqrt{x}$  обозначены соответственно  $g_\alpha(n), h_\alpha(n)$ .

2.1. Решены задачи, аналогичные п. 1.1—1.3, для  $g_\alpha(n)$  и  $h_\alpha(n)$ . В частности показано, что

$$h_1(n) = n[\sqrt{n}] - \frac{1}{3}[\sqrt{n}]^3 - \frac{1}{2}[\sqrt{n}]^2 + \frac{5}{6}[\sqrt{n}]. \quad (3)$$

2.2. Вычислены пределы:

$$а) \lim_{n \rightarrow \infty} \frac{g_\alpha(n)}{n^3} = \frac{\alpha}{3}; \quad б) \lim_{n \rightarrow \infty} \frac{h_\alpha(n)}{n^{\frac{3}{2}}} = \frac{2\alpha}{3}.$$

2.3. а) Доказаны формулы  $h_\alpha(n) = h_{[a]}(n) + h_{\{\alpha\}}(n)$  и  $g_\alpha(n) = \frac{[\alpha]n(n+1)(2n+1)}{6} + g_{\{\alpha\}}(n)$ .

б) Доказано, что, если  $\alpha$  - положительное иррациональное, то  $g_\alpha(n) + h_{\frac{1}{\alpha}}([n\alpha^2]) = n[\alpha n^2]$

2.4.

1) а) Доказано, что  $g_{\alpha+\beta}(n) = g_\alpha(n) + g_\beta(n)$  при  $\alpha\beta \geq 0$  и  $g_{\alpha+\beta}(n) < g_\alpha(n) + g_\beta(n)$  при  $\alpha \in \mathbb{Z}, \beta \in \mathbb{Z}, \alpha\beta < 0$ .

б) Если  $\alpha \geq 0$ , то  $g_\alpha(n+m) < g_\alpha(n) + g_\alpha(m)$  и  $g_\alpha(n+m) < g_\alpha(n) + g_\alpha(m)$  при  $\alpha < 0$ .

2) а) Доказано, что  $h_{\alpha+\beta}(n) = h_\alpha(n) + h_\beta(n)$  при  $\alpha\beta \geq 0$  и  $h_{\alpha+\beta}(n) < h_\alpha(n) + h_\beta(n)$  при  $\alpha \in \mathbb{Z}, \beta \in \mathbb{Z}, \alpha\beta < 0$ .

б) Если  $\alpha \geq 0$ , то  $h_\alpha(n+m) < h_\alpha(n) + h_\alpha(m)$  и  $h_\alpha(n+m) < h_\alpha(n) + h_\alpha(m)$  при  $\alpha < 0$ .

Также получены формулы для вычисления точек под следующими кривыми:

$$N = \sum_{x=1}^n [\alpha x^m]; \quad (4)$$

$$N = \sum_{x=1}^n [\alpha^x]; \quad (5)$$

$$N = \sum_{x=1}^n [\log_{\alpha} x]; \quad (6)$$

$$N = \sum_{x=1}^n \left[ \frac{\alpha \sin x + |\alpha \sin x|}{2} \right]; \quad (7)$$

$$N = \left[ \frac{\alpha \pi}{2} \right]. \quad (8)$$

для  $y = \alpha x^m$  (8),  $y = \alpha^x$  (9),  $y = \log_{\alpha} x$  (10),  $y = \alpha \sin x$  (11),  $y = \alpha \arcsin x$  (12).

В трехмерном пространстве для плоскости  $z = \alpha x + \beta y$  получено обобщение:

$$\sum_{y=1}^n (\sum_{x=1}^{n-y} [\alpha x + \beta y]) \quad (9)$$

Таким образом, в процессе исследования было рассмотрено количество натуральных точек под различными видами кривых, получены формулы для их вычисления и изучены свойства этих множеств.

**Список использованных источников:**

1. Ильин В.А., Садоничий В.А., Сендов Бл. Х. Математический анализ. Часть 1.
2. Ильин В.А., Садоничий В.А., Сендов Бл. Х. Математический анализ. Часть 2

## РАВЕНСТВО ПАРСЕВАЛЯ

*Якутонис Алексей Сергеевич*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Каянович Сергей Сергеевич – канд. физ.-мат. наук*

Равенство Парсеваля — это аналог теоремы Пифагора в векторных пространствах со скалярным произведением. Доказательство с помощью предельного перехода под знаком интеграла.

Равенство Парсеваля.

Степенной ряд

$$f(z) = \sum_{k=0}^{\infty} a_k (z - z_0)^k \quad (1)$$

Можно рассматривать как некоторый аналог ряда Фурье для функции  $f(z)$ . Чтобы в этом убедиться, докажем сначала лемму.

Теорема

$$\frac{1}{2\pi} \int_0^{2\pi} |f(z_0 + \rho e^{i\theta})|^2 d\theta = \sum_{k=0}^{\infty} |a_k| \rho^{2k} \quad (2)$$

Замечая, что  $|z|^2 = z\bar{z}$ ,  $\overline{z_1 z_2} = \bar{z}_1 \bar{z}_2$ ,  $\overline{z^m} = \bar{z}^m$ , получаем цепочку равенств

$$\begin{aligned} \frac{1}{2\pi} \int_0^{2\pi} \left| \sum_{k=0}^n a_k (z - z_0)^k \right|^2 d\theta &= \frac{1}{2\pi} \int_0^{2\pi} \left( \sum_{k=0}^n a_k (z - z_0)^k \cdot \overline{\sum_{j=0}^n a_j (z - z_0)^j} \right) d\theta = \\ &= \frac{1}{2\pi} \int_0^{2\pi} \left( \sum_{k=0}^n a_k (z - z_0)^k \cdot \sum_{j=0}^n \bar{a}_j \overline{(z - z_0)^j} \right) d\theta = \frac{1}{2\pi} \int_0^{2\pi} \left( \sum_{k,j=0}^n a_k \bar{a}_j (z - z_0)^k \overline{(z - z_0)^j} \right) d\theta = \\ &= \frac{1}{2\pi} \sum_{k,j=0}^n a_k \bar{a}_j \int_0^{2\pi} (z - z_0)^k \overline{(z - z_0)^j} d\theta = \frac{1}{2\pi} \sum_{k=0}^n a_k \bar{a}_k \int_0^{2\pi} (z - z_0)^k \overline{(z - z_0)^k} d\theta = \sum_{k=0}^n |a_k| \rho^{2k}, \end{aligned}$$

так как  $z - z_0 = \rho e^{i\theta}$ ,  $(z - z_0)\overline{(z - z_0)} = |z - z_0|^2 = \rho^2$ .

Итак, получено

$$\frac{1}{2\pi} \int_0^{2\pi} \left| \sum_{k=0}^n a_k (z - z_0)^k \right|^2 d\theta = \sum_{k=0}^n |a_k|^2 \rho^{2k}.$$

Поскольку последовательность

$$\left\{ \sum_{k=0}^n a_k (z - z_0)^k \right\}$$

на окружности  $C_\rho$  сходится равномерно, то и последовательность модулей

$$\left\{ \left| \sum_{k=0}^n a_k (z - z_0)^k \right| \right\}$$

будет равномерно сходящейся на это окружности. Это вытекает из неравенства

$$\left| \left| \sum_{k=0}^{n+p} a_k (z - z_0)^k \right| - \left| \sum_{k=0}^n a_k (z - z_0)^k \right| \right| \leq \left| \sum_{k=0}^{n+p} a_k (z - z_0)^k - \sum_{k=0}^n a_k (z - z_0)^k \right|$$

Следовательно, возможен предельный переход под знаком интеграла, в результате которого получаем:

$$\frac{1}{2\pi} \int_0^{2\pi} |f(z_0 + \rho e^{i\theta})|^2 d\theta = \lim_{n \rightarrow \infty} \frac{1}{2\pi} \int_0^{2\pi} \left| \sum_{k=0}^n a_k (z - z_0)^k \right|^2 d\theta = \lim_{n \rightarrow \infty} \sum_{k=0}^n |a_k|^2 \rho^{2k}.$$

Отсюда следует, что ряд  $\sum_{k=0}^n |a_k|^2 \rho^{2k}$  сходится и его сумма есть интеграл

$$\frac{1}{2\pi} \int_0^{2\pi} |f(z_0 + \rho e^{i\theta})|^2 d\theta,$$

т. е. выполняется равенство (2). Равенство (2) аналогично равенству Парсеваля в теории рядов Фурье.

**Список использованных источников:**

1. Привалов И.И. Граничные свойства аналитических функций: Рипол Классик, 2013 - С. 89-90.
2. Маркушевич А.И. Теория аналитических функций. Т. 2. Дальнейшее построение теории: ООО ДиректМедиа, 27 мая 2016 г. – С. 136-137.

## НЕЕКЛИДОВА ГЕОМЕТРИЯ, ТОПОЛОГИЯ

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Земнов Д.О., Чернявский И.В.*

*Минченков Ю.В. – кандидат физико-математических наук, доцент*

Неевклидова геометрия — в буквальном понимании — любая геометрическая система, которая отличается от геометрии Евклида; однако традиционно термин «неевклидова геометрия» применяется в более узком смысле и относится только к традиционным неевклидовым геометрическим системам: геометрии Лобачевского и геометрии Римана.

Как и евклидова, эти геометрии относятся к метрическим геометриям пространства постоянной кривизны. Нулевая кривизна соответствует евклидовой геометрии, положительная — совпадающим по локальным свойствам сферической или геометрии Римана, отрицательная — геометрии Лобачевского. На рисунке 1 изображено представление неевклидовой геометрии.



Представление неевклидовой геометрии

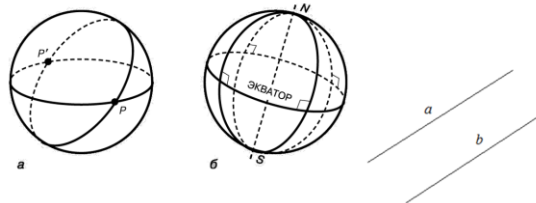


Рис. 1 – Представление неевклидовой геометрии

Геометрия Евклида говорит, что вы можете провести прямую линию. Прямая линия – это кратчайшее расстояние между двумя точками. Неевклидова геометрия меняет само направление научной мысли. Неевклидова геометрия утверждает, что прямую линию провести вообще невозможно. Почему? Потому что вы находитесь на Земле, которая представляет собой сферу. Если вы проводите линию, она кажется только прямой, поскольку вы забываете, что находитесь на земном шаре. Начните проводить линию, ведите и ведите, и вскоре вы увидите, что она превращается в круг, потому что покрывает всю Землю. А то, что прямая линия не может быть частью окружности, очевидно; если она является частью окружности, она не прямая. Никакая прямая линия не может создать окружность, однако любая прямая линия, которая проведена до самого конца становится частью круга. Но тогда это окружность, а не прямая. Вывод: вся вселенная состоит из окружностей. Всё движение во вселенной происходит по кругу; всё представляет собой круг. Прямые линии невозможны, это воображаемые линии. Так гласит неевклидова геометрия. Неевклидова геометрия включает в себя геометрию Лобачевского и геометрию Римана.

Топология тесно связана с неевклидовой геометрией. Топология — раздел математики, который изучает свойства фигур, сохраняющиеся при непрерывных деформациях — растяжении, сжатии или изгибании. Такие геометрические свойства связаны с положением, а не с формой или величиной фигуры. Иначе говоря, в топологии рассматривается, как превратить один объект в другой без разрывов и склеиваний и что при этом произойдет.

Для наглядности топологию часто представляют как «геометрию на резиновом листе». Если представить, что есть шар, сделанный из идеальной резины, непрерывными деформациями (без разрывов и склеек) его можно преобразовать в куб, конус или цилиндр, но не в тор. Топологически все эти фигуры эквивалентны, то есть гомеоморфны между собой. При помощи топологии был представлен ряд открытий, полезных обществу. К ним относятся исследования в квантовой химии, кристаллографии, непосредственно в математическом и функциональном анализе и непосредственно в математической физике.

Также топология, вероятно, способна ответить на вопрос - какой формы вселенная? Здесь топология встречается с астрономией. Форма вселенной может изгибаться, иметь положительную и отрицательную кривизну в разных её точках, о чем рассказывал мой коллега. То есть пространство может быть не только плоским, но и изгибаться и соответственно подчиняется правилам геометрии Лобачевского, сферическая геометрия.

Чтобы узнать, какой формы Вселенная, астрофизики измерили плотность вселенной по формуле  $E=mc^2$ , переведя энергию всего распространившегося света в массу. Орбитальный телескоп Планк измерил среднюю плотность вселенной, и, после вычислений, астрофизики пришли к выводу, что вселенная - плоская. Но это еще не значит, что вселенная не может быть замкнута сама на себя. Она может быть замкнута в тор, имеющий по сути одну плоскость. Но вселенная действительно может быть замкнута, только в четырехмерном пространстве.

**Список использованных источников:**

1. Александров П. С. Что такое неевклидова геометрия. — М.: УРСС, 2007.
2. Клейн Ф. Неевклидова геометрия.
3. Виро О. Я., Иванов О. А., Харламов В. М., Нецветаев Н. Ю. Элементарная топология – 2007.

## **СЕКЦИЯ «ИНФОРМАТИКА»**

# АЛГОРИТМ ПОДСЧЁТА РАЦИОНАЛЬНЫХ ТОЧЕК ВЛИЗИ ПАРАБОЛЫ

Андреюк Д.С., Труханович М.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Калугина М.А. – канд. физ.-мат. наук, доцент

В работе рассмотрены алгоритмы решения задачи распределения точек с рациональными координатами в некоторой окрестности параболы. Приведены подходы, лежащие в основе выбранной методики подсчета точек, и асимптотические оценки трудоемкости получения результата.

Задачи о количестве целых точек в областях евклидова пространства возникли в классической теории чисел. Важное место среди них занимает подсчет на плоскости точек с целочисленными координатами, которые лежат внутри замкнутой кривой [1]. Как обобщение возникла задача о распределении рациональных точек вблизи кривых, эффективное решение которой играет важную роль в получении оценок размерности Хаусфорда множеств, определяемых диофантовыми неравенствами при совместных приближениях [2].

Поскольку множество рациональных чисел бесконечно, в рамках поставленной задачи рассматриваются координаты, представимые несократимой дробью с целым числителем и натуральным знаменателем, не превышающим некоторого значения.

Выбранный способ подсчёта точек сводится к нахождению на открытом вещественном интервале  $(a, b)$  рациональных точек со знаменателем, не превышающим  $Q \in \mathbb{N}$ . Его можно рассматривать как обобщение подсчёта целочисленных координат на отрезке: число точек, представимых в виде рациональных дробей с натуральным знаменателем, равно числу целых чисел на этом отрезке, умноженном на вышеназванный знаменатель. Однако, ввиду основного свойства дроби, одно и то же рациональное число может быть задано дробями с разными знаменателями. Таким образом, возникает проблема нахождения именно несократимых дробей на интервале.

В ходе решения поставленной задачи был изучен ряд Фарея [3] – последовательность положительных несократимых правильных дробей со знаменателем, не превышающим порядок последовательности. Таким образом, если рассматриваемый интервал включает промежуток единичной длины с целочисленной точной нижней гранью, то задача поиска рациональных точек представляет собой построение ряда Фарея требуемого порядка с последующим смещением его элементов. Для промежутков, которые не могут быть покрыты единичными интервалами, необходимо применение наивного алгоритма перебора всех дробей с проверкой их на несократимость посредством нахождения НОД алгоритмом Евклида.

Таким образом, каждой рациональной координате на оси абсцисс можно поставить в соответствие некоторое число точек с рациональными ординатами. Следовательно, решение задачи нахождения рациональных точек на интервале оси абсцисс позволяет определить отрезки на окрестности рассматриваемой гладкой кривой (в случае данной работы -- параболы), параллельные оси ординат, на которых будут располагаться искомые рациональные точки. Проецирование этих отрезков на ось ординат и решение задачи нахождения рациональных точек на интервале для полученных промежутков в итоге предоставит решение рассматриваемой проблемы. Отдельной задачей является оценка быстродействия разработанных алгоритмов, что одновременно позволяет найти приближение числа искомых точек.

Таким образом, можно выделить два алгоритма: подсчёт рациональных точек на интервале и непосредственное определение их координат. Оба алгоритма подразумевают фрагментацию рассматриваемого интервала, построение ряда Фарея заданного порядка, перебор значений на концевых промежутках заданного интервала, не покрываемых единичными отрезками. Учитывая рекурсивность алгоритма построения ряда Фарея  $F_n$  и приближение числа элементов последовательности  $n$ -го порядка вида

$$|F_n| \sim \frac{3n^2}{\pi^2},$$

а также логарифмическое время работы алгоритма Евклида, асимптотически сложность алгоритма подсчёта рациональных точек на интервале  $(a, b)$  со знаменателем, не превышающим  $Q \in \mathbb{N}$ , можно описать как

$$O(Q^3 + (b-a)Q^2 \log(Q)),$$

в то время как быстродействие алгоритма нахождения рациональных точек с теми же входными данными может быть описано как

$$O(Q^3 + (b-a)Q^2(\log(Q) + 1)),$$

поскольку требуется проведение операций смещения ряда Фарея.

В рамках описываемого алгоритма нахождение числа рациональных точек в окрестности параболы требует определения рациональных абсцисс. Однако дальнейшие этапы алгоритма используют результаты этого определения. Потому оценка сложности требует нахождения приближения числа рациональных точек на интервале. Это может быть осуществлено с использованием теоремы о распределении простых чисел:

$$\lim_{n \rightarrow \infty} \frac{\pi(n)}{n / \ln n} = 1,$$

где  $\pi(n)$  -- функция распределения простых чисел. Предпосылкой этого служит факт, что взаимная простота числителя и знаменателя является критерием несократимости дроби. На этом основании получаем оценку числа рациональных точек на вещественном интервале  $(a, b)$  со знаменателем, не превышающим  $Q \in \mathbb{N}$  :

$$O\left(\frac{1}{2}(b-a)(Q^2 + Q) - 2Q + 1 - \sum_{i=1}^Q \frac{(b-a)i-1}{\ln[\sqrt{i} + 1]}\right),$$

которая, однако, требует нахождения частичной суммы указанного ряда, что является нетривиальной задачей. Альтернативой является использование оценки количества элементов ряда Фарея, что даёт асимптотическую оценку:

$$O((b-a)Q^2).$$

Таким образом, быстродействие алгоритма по подсчёту рациональных точек в  $\varepsilon$ -окрестности параболы на вещественном интервале  $(a, b)$  со знаменателем, не превышающим  $Q \in \mathbb{N}$  может быть оценено как

$$O((b-a)Q^5 + \varepsilon(b-a)Q^4 \log(Q)),$$

в то время как для алгоритма нахождения искомым точек с теми же входными данными справедлива оценка:

$$O((b-a)Q^5 + \varepsilon(b-a)Q^4 \log(Q+1)).$$

В исследовании В. Бересневича [2] рассматривался случай  $\varepsilon = Q^{-\nu-1}, 0 < \nu < 1$ . При подстановке несложно сравнить оба результата.

Для этой же окрестности произвести оценку числа рациональных точек можно и на основании теоремы о распределении простых чисел:

$$O\left(\left(\frac{b-a}{2}(Q^2 + Q) - 2Q + 1 - \sum_{i=1}^Q \frac{(b-a)i-1}{\ln[\sqrt{i} + 1]}\right)(Q^{-\nu-1}(Q^2 + Q) - 2Q + 1 - \sum_{i=1}^Q \frac{2Q^{-\nu-1}i-1}{\ln[\sqrt{i} + 1]}\right)),$$

и на основании свойств ряда Фарея:

$$O\left(\frac{9}{\pi^2}(b-a)Q^{-\nu+3}\right).$$

Разработанные алгоритмы были реализованы на языке программирования Python 3.0 с использованием библиотек NumPy и Matplotlib. Демонстрация работы для интервала  $(-1.01; 2.99)$  с верхним ограничением на знаменатель  $Q = 5$  в окрестности  $\varepsilon = 1$  представлена на рисунке 1.

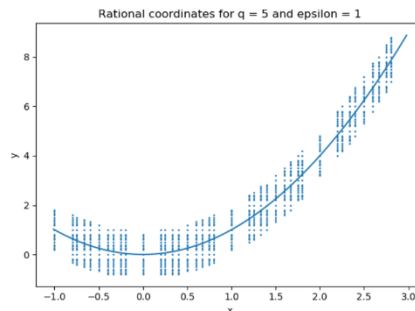


Рисунок 1 – Распределение рациональных точек вблизи параболы

**Список использованных источников:**

1. Бухштаб, А.А. Теория чисел / А.А. Бухштаб. – Москва, издательство «Просвещение», 1965. -- 51 с.
2. Бересневич, В.В. Распределение рациональных точек вблизи параболы / В.В. Бересневич // Материалы докладов

## АЛГОРИТМ АВТОМАТИЧЕСКОГО ОПИСАНИЯ ИЗОБРАЖЕНИЙ

*Астрашав В.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Калугина М.А. – канд. физ.-мат. наук, доцент*

В докладе рассмотрен алгоритм автоматического описания изображений, работающий на основе нейронных сетей. В частности, приведено общее описание принципов его реализации, проанализированы полученные результаты и перспективы его применения.

Автоматическая генерация описаний для изображений является важной задачей, успешное решение которой позволит сэкономить человеческие ресурсы. Правилom хорошего тона является сопровождать все изображения, размещённые на веб-сайтах, текстовым описанием. Это делается по нескольким причинам. Во-первых, это упрощает пользование интернетом и доступ к информации людям с проблемами со зрением, так как текстовая информация может быть озвучена голосовым помощником, а изображения зачастую содержат важную для понимания сути написанного информацию. Во-вторых, этот текст может быть отображён вместо изображения если возникли проблемы с его загрузкой. На данный момент большая часть таких описаний пишется вручную или вовсе игнорируется необходимость их написания. Но на некоторых сайтах, таких как новостные ресурсы, интернет-энциклопедии и другие подобные сайты, изображений содержится так много, что на их описание тратится много времени авторов статей. Кроме того, необходимость описывать изображения возникает в учебниках, технической литературе, научных статьях. Алгоритмы автоматической генерации описаний помогли бы сэкономить время авторам. Даже если качество описаний не будет таким высоким, как у человека, перепроверка и корректирование автоматически сгенерированного описания займут меньше времени, чем составление собственного с нуля.

В данной работе рассмотрен один из современных алгоритмов генерации описаний к изображениям, разработанный в 2016 году исследователями университетов Монреаля и Торонто [1]. Данный алгоритм основан на использовании технологии больших данных. Работу по обработке изображения и перевода его содержания в текст совершают несколько нейронных сетей, предварительно обученных для этого на наборе данных, содержащим множество примеров изображений и описаний к ним. Плюсом этого подхода является то, что программисту не нужно знать особенностей синтаксиса и грамматики языка, уметь выделять из изображения необходимые признаки вручную и сопоставлять их со словами. Все эти задачи нейросетевая модель учится совершать самостоятельно с помощью алгоритмов оптимизации. Задача, выполнение которой остаётся за программистом, – это проектирование архитектуры модели, её реализация на одном из языков программирования, а также сбор и подготовка необходимого набора данных. Последний пункт долгое время был наиболее трудным в осуществлении, так как сложные модели для качественного обучения требуют огромного набора данных, а также ресурсов для их хранения и обработки. В последнее время разработки в сфере больших данных сделали эту задачу осуществимой, поэтому алгоритмы такого типа завоевывают популярность в сравнении с менее требовательными к данным, но и менее точными и более сложными в разработке классическими алгоритмами.

Реализованная нами модель состоит из трёх нейронных сетей: кодирующей нейронной сети, декодирующей нейронной сети, а также нейронной сети с вниманием.

Задача кодирующей сети состоит в переводе изображения в сжатый вектор признаков, по которым в дальнейшем будет осуществляться составление описания. В качестве кодирующей сети можно брать любую архитектуру для работы с изображением, например, наиболее популярные свёрточные нейронные сети. Для данной архитектуры была выбрана сеть ResNet-101 [2], состоящая из 101 свёрточного слоя. Эта сеть уже была настроена классифицировать изображения, поэтому умеет выделять некоторые признаки, но ее можно дополнительно обучить решать конкретную задачу. Такой подход позволяет сети обучаться быстрее, так как не приходится делать это с нуля.

Задача декодирующей сети состоит соответственно в генерации текста по выделенным признакам. Для этого используется рекуррентная нейронная сеть с LSTM-ячейками памяти [3]. Получая вектор признаков, она генерирует по нему слово и некоторый вектор скрытого состояния. Этот вектор она передаёт в точно такую же ячейку. Процедура повторяется до тех пор, пока сетью не будет сгенерирован символ конца предложения. Также для генерации каждого слова декодирующая сеть принимает на вход результат работы сети с вниманием.

Сеть с вниманием работает совместно с декодирующей сетью. Она получает на вход предыдущее сгенерированное слово, а также признаки, выделенные кодирующей сетью. Её задача заключается в том, чтобы выделить наиболее важные из признаков для генерации следующего слоя. Это позволяет модели «обращать внимание» на различные признаки попеременно, наподобие того, как человек рассматривает изображение, последовательно замечая различные объекты на нём. Данная сеть была предложена исследователем Монреальского университета, выпускником факультета прикладной математики и информатики БГУ Дмитрием Богдановым [4]. Изначально эта архитектура использовалась для задач машинного перевода, но впоследствии нашла своё применение и в других задачах.

Все нейронные сети из приведённой архитектуры затем обучаются совместно. Обучение проводилось на наборе данных MSCOCO '14, содержащим более 200 тысяч размеченных изображений и находящимся в открытом доступе для использования в исследовательских целях. Каждое изображение из данного набора данных имеет своё описание на английском языке. Каждое изображение было сжато нами до 256x256 пикселей. Общий вес набора данных составляет около 19 Гб.

Обучение проходит на тренировочном наборе данных, а проверка регулярно осуществляется на наборе данных для валидации. Это позволяет отследить эффект переобучения, когда модель слишком сильно приспосабливается к тренировочным данным и теряет способность обобщения на новые данные. Это можно заметить с помощью валидации, если во время обучения качество модели на тренировочных данных растёт, а на валидационных – убывает. Для предотвращения проблемы переобучения используется регуляризация. В данной модели используется один из способов регуляризации – накладывание ограничения на величину параметров модели.

Для обучения используются алгоритмы оптимизации, а оптимизируется функция потерь, в качестве которой используется перекрёстная энтропия.

Для реализации данной модели мы использовали язык программирования Python 3.6 с библиотекой для глубокого обучения TensorFlow и множеством вспомогательных библиотек. Выбор данного языка и библиотек обусловлен наличием в них наиболее широкого набора инструментов для разработки моделей машинного обучения.

Пример результатов работы алгоритма приведён на рисунке 1. Над изображением содержится описание, сгенерированное с помощью алгоритма на английском языке.

Prediction Caption: a male surfer riding the waves in the ocean <end>



Рисунок 1 – пример работы модели на изображении

Таким образом, в результате проведённой работы, был изучен и реализован алгоритм автоматической генерации описания к изображениям. При наличии соответствующего набора данных сфера применения данного алгоритма может быть расширена. В частности, ту же самую архитектуру можно использовать для автоматического составления описаний на любых языках, а не только на английском. Набор данных может быть заменён на более подходящий к специфичной области применения для улучшения работы алгоритма на изображениях в данной области. Если такой набор данных достаточно велик и существенно отличается от предложенного в работе, можно провести обучение модели с нуля, иначе можно использовать уже обученную модель и дообучить её для более конкретной задачи. Работает данный алгоритм с большой скоростью (менее секунды на одно изображение), что позволяет быстро генерировать описание для множества изображений.

**Список использованных источников:**

1. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention / K. Xu [et al.] // arXiv:1502.03044v3 [cs.LG], 2016.
2. Deep Residual Learning for Image Recognition / K. He [et al.] // arXiv:1512.03385v1 [cs.CV], 2015.
3. Long Short-term Memory / S. Hochreiter, J. Schmidhuber // Neural Computation 9(8):1735-80, 1997
4. Neural Machine Translation by Jointly Learning to Align and Translate / D. Bahdanau [et al.] // arXiv:1409.0473v7 [cs.CL], 2016

## **ОБЗОР МЕТОДОВ ИДЕНТИФИКАЦИИ ФЛЕШ-НАКОПИТЕЛЕЙ С ИСПОЛЬЗОВАНИЕМ ФИЗИЧЕСКИ НЕКЛОНИРУЕМЫХ ФУНКЦИЙ**

*Белясова А. В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Заливако С. С. – кандидат техн. наук*

В данной работе рассматриваются некоторые из существующих методов идентификации, реализованных на основе физически неклонированных функций, для устройств на основе NAND флеш-памяти. Приводится экспериментальная оценка основных характеристик методов, таких как производительность, стабильность и уникальность. Все методы основаны на вычислении выходных значений ФНФ и применении алгоритма генерации идентификатора для устройства. Предложены возможные направления работы с целью улучшения производительности методов.

В сфере производства товаров актуальной проблемой является их нелегальное копирование. За последние 10 лет количество товаров-подделок увеличилось в 2 раза. Производители несут серьезные убытки, что характерно в том числе для рынка электроники [1]. Чтобы защитить свои товары, производители прибегают к таким методам, как аппаратное шифрование, хеширование, внедрение водяных знаков и другие. Однако многие из этих методов требуют значительных аппаратных затрат, которые сопоставимы с затратами на само устройство. На данный момент одним из менее требовательных к аппаратуре методов являются физически неклонированные функции (ФНФ) [1]. В данной работе рассматриваются ФНФ, использующие уникальность пороговых напряжений для различных состояний элементов NAND флеш-памяти [2] и ошибки, возникающие в страницах, смежных с записываемыми.

Первый метод реализации ФНФ называется методом частичного стирания (partial erasure). Его алгоритм в следующем. Стиранию подвергается выбранный блок, затем выбранная страница записывается «нулями». Выполняется фиксированное число итераций частичного стирания выбранного блока [3]. Пороговые напряжения разные для элементов памяти, поэтому только некоторые из них поменяют состояние на '1'. Этим элементам присваивается номер итерации стирания. После последней итерации всем нестертым элементам памяти будет присвоено число, равное номеру следующей итерации. Значения, присвоенные каждому из элементов, являются выходными значениями ФНФ.

Второй метод – метод частичной записи (partial programming). Алгоритм аналогичен методу частичного стирания, за исключением того, что выполняются итерации частичного программирования, а не стирания. Также вначале, после стирания блока, запись в выбранную страницу не происходит. Операция частичного программирования имеет фиксированный период времени выполнения.

Третий метод называется методом внесения ошибок в результате записи (program disturbance). Алгоритм производит стирание выбранного блока и фиксированное число записей в выбранную страницу. На каждой итерации проверяется состояние битов соседней страницы. В результате каждому биту присваивается число, равное номеру итерации, на которой он изменил состояние, или номеру итерации, следующей за последней, если бит так и не был запрограммирован. Главным недостатком данного метода является его применимость только на 2D NAND (при плоской организации флеш-памяти). На более современном 3D NAND помехи в соседних страницах при чтении/записи встречаются существенно реже.

После получения выходных значений ФНФ необходимо применить алгоритм генерации идентификаторов. Часто для извлечения идентификаторов используется метод нечеткого экстрактора, которые используют криптографическую хеш-функцию и коды коррекции ошибок (ECC) для повышения стабильности, что требует дополнительных аппаратных затрат. В работе [3] описываются более экономичные методы генерации идентификатора, не требующие дополнительных затрат, такие как bit-map и position-map. Каждый метод генерирует идентификаторы длины  $X$  из выходных значений ФНФ для последовательности ячеек (элементов памяти) длины  $Y$ , где  $Y \geq X * 2$ .

В bit-map методе ячейки разбиваются на пары и вычисляется разность значений ФНФ в каждой

паре. Выбирается  $X$  пар с максимальной разностью. В position-map методе ячейки сортируются по значению ФНФ. Ячейки с минимальным и максимальным значениями образуют пару и исключаются из рассмотрения. Таким образом выбираются  $X$  последовательных пар. В паре ячейки сортируются по адресу в памяти. На основе выбранных пар генерируется идентификатор. Если первая ячейка в паре имеет большее значение ФНФ, чем вторая, то соответствующий бит идентификатора равен единице, иначе нулю.

Результаты экспериментов в работе [3] показали, что производительность для метода частичного стирания на разных устройствах составила от 12,21 до 14,67 Кбит/с, для метода частичного программирования – от 14,45 до 22,38 Кбит/с, для метода внесения ошибок в результате записи – от 7,35 до 9,72 Кбит/с. Это позволило генерировать 128-битные идентификаторы за время от 93,83 мс до 2,48 с.

Метод частичного программирования оказался быстрее метода частичного стирания, так как период частичного стирания был выбран на порядок больше периода частичного программирования для всех исследованных чипов NAND при практически одинаковых значениях количества итераций методов (например, 245 и 250 соответственно). Метод внесения ошибок в результате записи оказался самым медленным, из-за того что количество его итераций было на порядок больше количества итераций предыдущих методов (3000-4000) и для его реализации использовалась полная операция программирования, длительность которой больше длительностей операций частичного стирания и частичного программирования [3].

Для оценки характеристики уникальности описанных ФНФ было вычислено расстояние Хэмминга между 10000 парами 128-битных идентификаторов. Данная случайная величина должна была иметь биномиальное распределение с  $N=128$  и  $p=0,5$  [3]. Матожидание величины в таком случае равно  $N/2$ , что значит, что различие идентификаторов должно быть 50%-ным в среднем. Эксперимент дал следующие результаты: 49,93% для метода частичной записи, 49,95% для метода частичного стирания и 46,86% для метода внесения ошибок в результате записи. Все методы дали хороший результат: отклонение от желаемых 50% составило не более 5% для каждого метода.

В работе [3] были произведены оценки стабильности посредством вычисления *отклонения*. *Отклонение* вычислялось как доля несовпавших битов. Было проанализировано, как зависит стабильность от количества выбранных для генерации битов. Пусть  $Y$  - количество генерируемых битов,  $X$  - количество выбранных для генерации битов,  $Y/X$  - их отношение (его максимальное значение - 0,5 [3]). Результаты экспериментов показали, что при уменьшении соотношения  $Y/X$ , уменьшается *отклонение*. Для bit-map метода уменьшение от 0,5 до 0,055 привело к уменьшению *отклонения* от 14,42% до 0,0001% при любых изменениях факторов окружающей среды. Для position-map метода уменьшение от 0,5 до 0,474 привело к уменьшению *отклонения* от 0,023% до 0,0001%.

Эталонным *отклонением* в работе [3] считалось отклонение 0,0001%, или  $10^{-6}$ . Это значит, что в одном из 1000000 случаев генерации битов произойдет ошибка. Таким образом, при генерации 128-битного идентификатора ошибка произойдет примерно один раз из  $1000000/128=7812,5$  попыток генерации. При этом вероятность ошибки в идентификаторе - около 0,000128. Если это недостаточно надежно, можно вычислить идентификатор повторно. При этом вероятность двойной ошибки значительно ниже ( $0,000128^2$ ).

Рассмотренные методы идентификации устройств могут быть успешно использованы для защиты авторских прав. Для их реализации не нужна дополнительная аппаратура, что позволяет использовать эти методы для устройств с ограниченными ресурсами (например, устройства Интернета вещей). Методы генерируют уникальные и стабильные идентификаторы, что позволит им также найти применение в генерации закрытых ключей для алгоритмов шифрования, тем самым устраняя необходимость в хранении секретных ключей непосредственно в памяти.

Главный недостаток рассмотренных методов - низкая производительность. Основная причина больших задержек в работе данных методов - многочисленное программирование/стирание выбранных страниц/блоков. Одним из последующих направлений деятельности может стать разработка метода, который предварительно вычисляет выходные значения ФНФ и компактно сохраняет данные ФНФ в память (NAND). Затем данные считываются и предоставляются алгоритму генерации идентификатора. Данный подход позволит вынести все самые длительные операции на этап предварительной обработки, что предположительно даст прирост в производительности при генерации идентификатора. Узким местом данного подхода является считывание данных ФНФ из NAND. Время чтения страницы данных размером 4 килобайта из NAND составляет от 25 до 100 мкс [4]. При максимальном количестве данных, необходимых каждому из рассмотренных методов для генерации идентификатора, одной страницы размером 4 килобайта предположительно хватит для их хранения. Таким образом, одного чтения будет достаточно для генерации идентификатора, что может позволить разработать метод значительно быстрее, чем 1-2 секунды.

#### Список использованных источников:

1. Физически неклонированные функции: защита электроники от нелегального копирования [Электронный ресурс]. — Режим доступа: <https://habr.com/ru/post/343386/>. — Дата доступа: 23.03.2020.



2. Understanding Flash: Floating Gates and Wear [Electronic resource]. — Mode of access: <https://flashdba.com/2015/01/09/understanding-flash-floating-gates-and-wear/>. — Date of access: 25.03.2020.
3. Extracting Robust Keys from NAND Flash Physical Unclonable Functions / J. Shijie, X. Luning, L. Jingqiang, J. Yafei // CCA Secure PKE with Auxiliary Input Security and Leakage Resiliency, 2015. — P. 437-454.
4. Coding for SSDs – Part 2: Architecture of an SSD and Benchmarking [Electronic resource]. — Mode of access: <http://codecapsule.com/2014/02/12/coding-for-ssds-part-2-architecture-of-an-ssd-and-benchmarking/>. — Date of access: 26.03.2020.

## ИСПОЛЬЗОВАНИЕ ГЕНЕРАТОРОВ КОНСТАНТ ДЛЯ ФОРМИРОВАНИЯ НЕОПТИМИЗИРУЕМЫХ VHDL ОПИСАНИЙ

*Видничук В.Н.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Иванюк А.А. – доктор технических наук*

Рассматриваются генераторы констант как один из методов внутрисхемной обфускации за счёт формирования не оптимизируемых VHDL описаний аппаратуры.

В связи с быстрым ростом производства аппаратуры и цифровых устройств растут и убытки связанные с пиратством. По последним данным ущерб составляет 17 млрд долларов в год. Также данная проблема влияет и на безопасность и надежность данных устройств. Следует выделить следующие угрозы: пиратство, внедрение аппаратной прослушки, аппаратные трояны, пиратство, реверс инжиниринг. Для борьбы с этими угрозами разрабатываются различные методы. Одним из главных методов – обфускация.

Обфускация это один из методов запутывания понимания функционирования схемы с целью защиты от обратного проектирования. Сложность и затраты на обратное проектирование схемы должно стремиться к максимальному значению и должны быть сопоставимы с затратами на разработку схемы с нуля. Также обфускацию можно использовать для внедрения водяных знаков.

В случае обфускации VHDL описаний рассматривают следующие виды обфускации: лексическая и функциональная.

Лексическая обфускация это изменение исходных проектных описаний для затруднения понимания их злоумышленниками, основным недостатком которой является то, что результат синтеза такого описания не изменится и при обратном проектировании злоумышленник может получить необфусцированное проектное описание[1].

Функциональная обфускация используется для получения более сложных схем путём применения функционально эквивалентных описаний. Функционально эквивалентное описание это описание которое по своей логике работает также но отличается по результатам синтеза. Основной проблемой при таком описании является оптимизатор. Оптимизация это процесс минимизации логических уравнений и приведения их к их минимальным формам. Проблема заключается в том что если оптимизатор видит одинаковые таблицы истинности то он приводит эти функции к одинаковым и результат синтеза(схема) будет одинаковой. Одним из решений проблемы может быть замена константных '0' и '1' генераторы констант. Так как оптимизатор не будет знать точную входную таблицу истинности он не сможет оптимизировать описание.

Генератор констант это разновидность непрозрачных предикатов, значение которых известны на этапе обфускации но требуют вычисления при анализе[2]. Они заменяют значения '0' и '1' что во время синтеза и оптимизации присоединяется к GND и VCC на схему сочетающую в себе последовательную и комбинационную логику. Данная схема разработана таким образом, чтобы генерировать константный '0' или '1' но при этом выглядеть как схема с памятью. Пример такой схемы можно увидеть на рис. 1.

На данном рисунке видно, что константный '0' заменён на схему с памятью, которая при любых входных значениях будет генерировать '0'. Генераторы констант можно использовать для замены простых логических примитивов на мультиплексор, повторяющий логическое уравнение данного примитива. Пример такой замены логического примитива AND представлен на рисунке 2. Где U1 это генератор константного '0' а U2 мультиплексор.

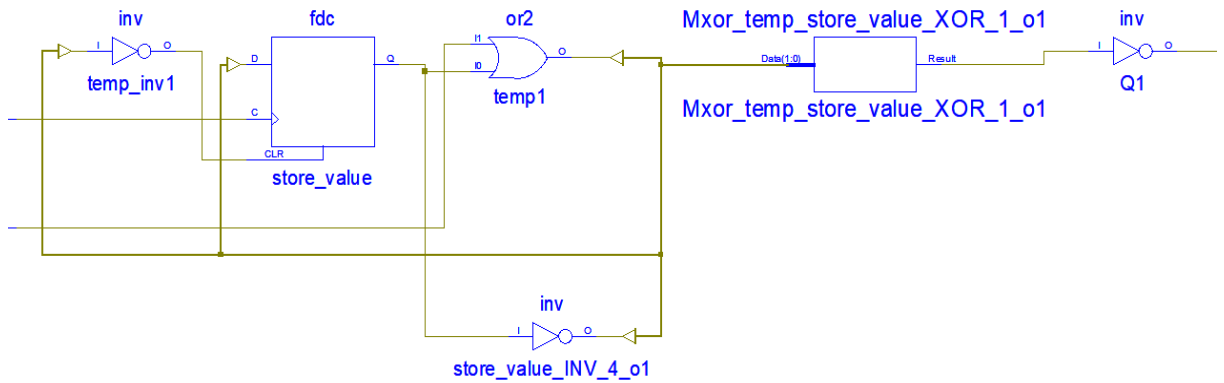


Рисунок 1 – схема не оптимизируемого генератора константного '0'.

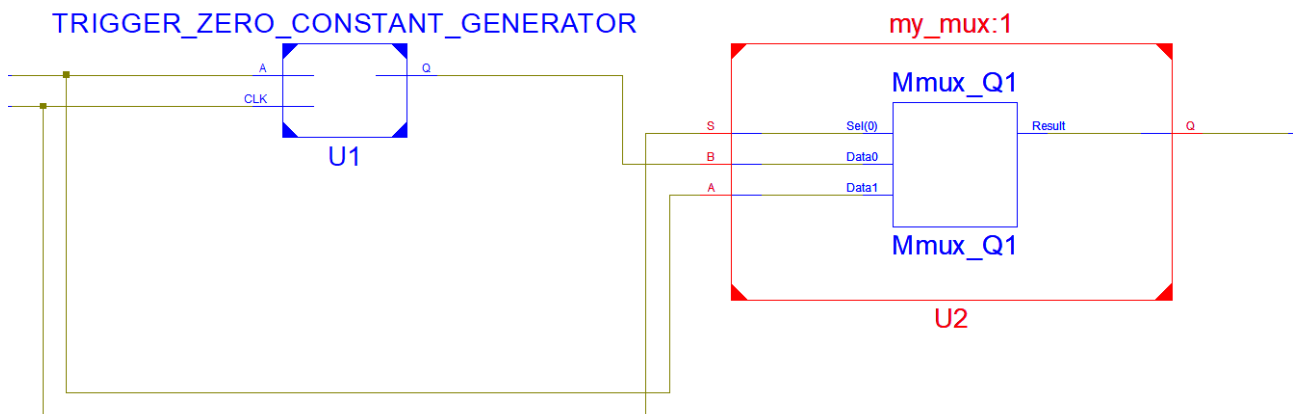


Рисунок 2 – пример схемы обфусцированного логического примитива AND с использованием генератора констант и мультиплектора.

Данная схема является не оптимизируемой из за того, что оптимизатор думает на выходе из генератора констант может выйти не '0' значение. Если использовать обычный константный ноль то оптимизатор минимизирует логическое выражение и на выходе будет обычный AND. На рисунке 3 показан результат технологического синтеза данного описания.

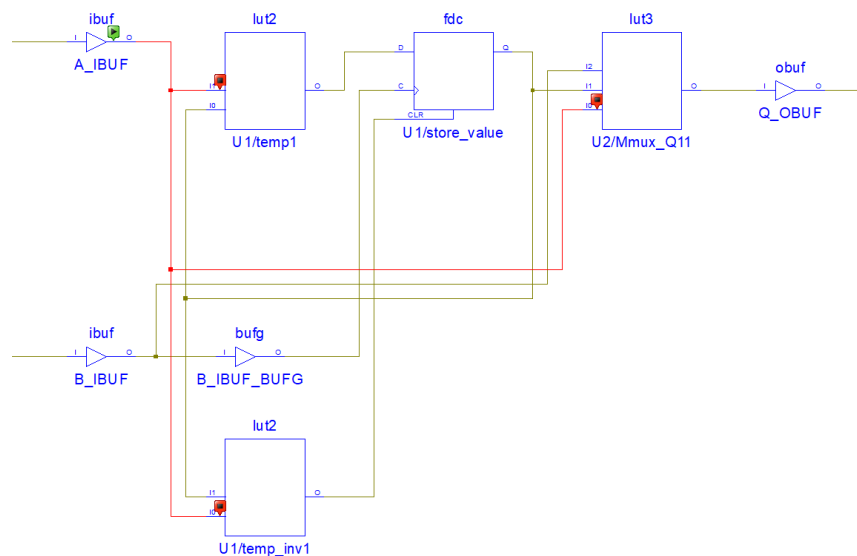


Рисунок 3 – результат технологического синтеза обфусцированного описания AND.

Результат синтеза подтверждает возможность использования функционально эквивалентных описаний для обфускации и помогает усложнить понимание схем, однако он имеет свои недостатки, например в связи с ростом количества элементов возрастают и общие задержки на схеме. Для увеличения сложности понимания имеет смысл использовать лексическую и функциональную обфускацию вместе.

**Список использованных источников:**

1. Видничук, В. Н. Лексическая обфускация как способ внедрения водяных знаков в исходные коды программ и проектных описаний / Видничук В. Н., Иванюк А. А. // Информационные технологии и системы 2019 (ИТС 2019) = Information Technologies and Systems 2019 (ITS 2019) : материалы международной научной конференции, Минск, 30 октября 2019 г. / Белорусский государственный университет информатики и радиоэлектроники; редкол. : Л. Ю. Шилин [и др.]. – Минск, 2019. – С. 136 – 137.
2. Сергейчик, В. В. Генераторы констант как базовые примитивы схемной обфускации / В. В. Сергейчик // Компьютерные системы и сети: материалы 50-й научной конференции аспирантов, магистрантов и студентов (Минск, 24-28 марта 2014 г.). – Минск : БГУИР, 2014. – С. 78 - 79.

## ИСПОЛЬЗОВАНИЕ СИСТЕМЫ КОМПЬЮТЕРНОЙ АЛГЕБРЫ MAPLE В ДИФФЕРЕНЦИАЛЬНОЙ ГЕОМЕТРИИ

*Гончарова А.В., Быстрова М.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Теслюк В.Н. – канд. физ.-мат. наук*

В данной работе представлены результаты изучения средств и пакетов программы системы компьютерной алгебры Maple и их использования для визуализации решений математических задач. Задачи, для которых созданы процедуры решения: построение однополостного гиперболоида с винтовой линией и вычисление её кривизны и кручения, построения графика поверхности, заданной параметрически, построение различных объёмных тел.

На сегодняшний день есть несколько популярных компьютерных инструментов, которые широко используются в математическом образовании, позволяют студентам достичь высокого уровня логико-аналитического мышления, визуально представляют сложную математическую информацию и рассматривают доказательства с помощью графических выражений.

В качестве иллюстрации мы выбрали математическую систему Maple.

Приведем наиболее яркие примеры решения задач дифференциальной геометрии с помощью данного пакета.

Одной из задач, наглядность которой является важной частью для понимания процесса решения, является задача исследования кривых. В качестве примера ниже представлен процесс вычисления кривизны и кручения винтовой линии на однополостном гиперболоиде с визуализацией и графическим представлением конечных формул. Для решения задачи рассмотрим однополостной гиперболоид вращения:  $ax = a * ch(u) * cos(v), y = a * ch(u) * sin(v), z = a * sh(u)$  и построим на нём винтовую линию с помощью уравнений  $u = \alpha * t, v = \beta * t$ . В Maple это будет выглядеть следующим образом:

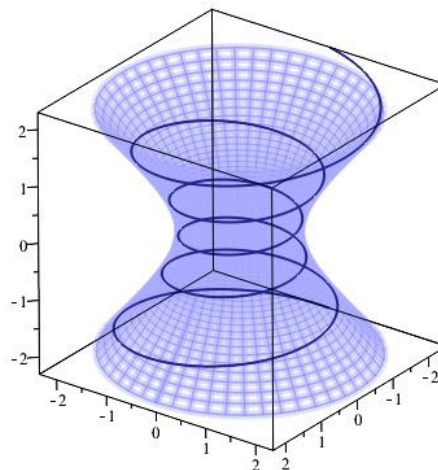


Рисунок 1 – Спираль на гиперболоиде вращения

Теперь, вычислив кривизну и кручение этой линии с помощью средств Maple, дадим формулам графическое представление:

Другим не менее наглядным примером удобства Maple будет построение графика поверхности, заданной параметрически. На рисунке 3 представлен вид поверхности, заданной параметрами  $x = 2 * u + v, y = v * cos(u), z = v * sin(u)$ .

Если учащиеся еще могут запомнить канонические уравнения поверхностей, их внешний вид,

то изобразить тело, ограниченное известными поверхностями, могут далеко не все. С этой задачей легко справиться с помощью программы Maple. На рисунке 4 представлено тело из сборника задач Кузнецова Л.А. [1], объем которого можно вычислить при помощи программного пакета Maple.

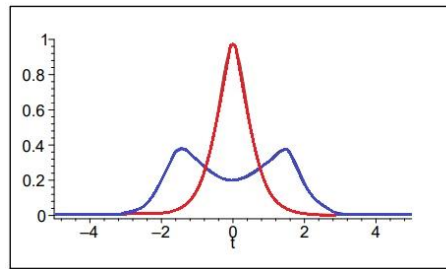


Рисунок 2 – Графики кривизны и кручения винтовой линии

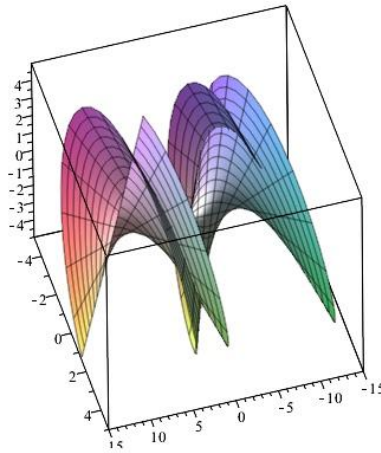


Рисунок 3 – График поверхности, заданной параметрически

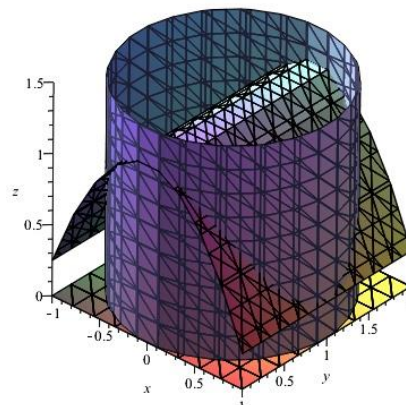


Рисунок 4 – Пример объемного тела

Ниже представлены ещё несколько интересных примеров тел, реализованных с помощью функций Maple:

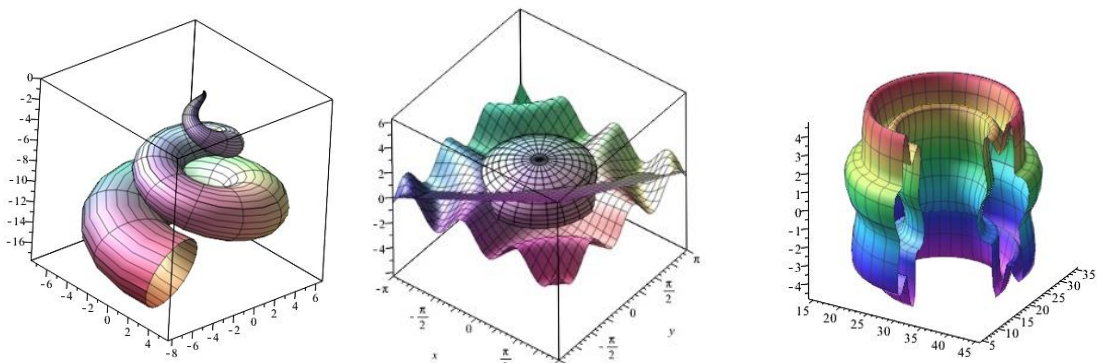


Рисунок 5 – Демонстрация других примеров построения тел в среде Maple

Приведённые выше примеры показывают, что средства Maple позволяют студенту получить наглядные результаты решения математических задач, прилагая меньше усилий, чем при вычислениях вручную.

**Список использованных источников:**

1. Кузнецов Л.А. Сборник заданий по высшей математике. Типовые расчеты: Учеб. пособие / Л.А. Кузнецов. 7-изд., стер. СПб.: Лань, 2005. – С. 141.
2. Касюк С.Т., Логвинова А.А. Высшая математика на компьютере в программе Maple 14. – Челябинск.: Изд. центр ЮУрГУ, 2011 – 57 с.
3. Maplesoft [Электронный ресурс]. – Режим доступа: <https://www.maplesoft.com/support/help/>

## ДЕТЕКТИРОВАНИЕ ВОЗГОРАНИЙ ПО ВИДЕОПОТОКУ

Дикий А.О.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Егорова Н.Г. – к.т.н., доцент

Время не обнаруженные и не ликвидированные возгорания вызывают большой экономический, экологический и людской ущерб. Разработка средств раннего обнаружения пожаров и задымлений имеет несколько направлений: использование физических или химических детекторов, видеокамер и алгоритмов компьютерного зрения, различных архитектур глубоких нейронных сетей или комбинация перечисленных средств вместе.

В классе задач машинного обучения, нацеленных на обработку изображений, широко используются сверточные нейронные сети. Они отличаются высокой робастностью, автоматическим выделением признаков, меньшим, по сравнению с классическими полносвязными сетями, количеством весов, возможностью использования GPU для ускорения процесса обучения, возможностью их переиспользования для новых задач.

Существует множество готовых решений для задач анализа и детектирования объектов, однако в данном исследовании выбор был сделан в пользу Faster R-CNN, поскольку она имеет более высокую скорость работы и точность, по сравнению с обычной R-CNN, а также обычно имеет более высокий показатель mean Average Precision (mAP), по сравнению с SSD и YOLO [1].

Faster R-CNN предоставляет выходы в виде ограничивающих рамок для классов огня, дыма и негорящих объектов. Негорящие объекты включают в себя объекты, которые имеют часть свойств пожаров или просто похожи на них. Использование более сложных алгоритмов, позволяющих проводить сегментацию изображений и более четкое выделение огня и дыма на изображении, нерационально в силу постоянной изменчивости объектов детектирования и высокой ресурсозатратности подобной операции.

Для решения обсуждаемой проблемы сверточные сети прекрасно подходят, однако одного детектирования изображения огня или дыма недостаточно, чтобы определить: имеется у нас возгорание или нет. Можно исключить несколько последних слоев, отвечающих за процесс детектирования в сверточной сети, и полученный вектор признаков передать в рекуррентную сеть, в частности в LSTM [2]. LSTM и GRU показывают схожие показатели эффективности работы, поэтому выбор архитектуры рекуррентной сети следует выбирать опытным путем. Стоит отметить, что для обучения рекуррентной сети в описываемой задаче не требуется большого набора тренировочных данных, достаточно порядка 50-100 видео с возгораниями и без них. Также можно попробовать взять предобученную на ImageNet VID рекуррентную сеть и дообучить на видео с пожарами. Однако в данной задаче этого не потребовалось, т.к. использование признаков из CNN модуля сильно упрощает работу для RNN, оно имеет только уточняющий предыдущий модуль эффект.

Рекуррентная сеть выдает ответ на вопрос: был ли детектирован пожар на коротком интервале времени. Иногда этого недостаточно, и требуется добавить дополнительный модуль, который будет сравнивать количество положительных и отрицательных выходов из RNN-модуля за определенное количество периодов. Кроме того, можно использовать детектированные в CNN модуле регионы пламени и дыма, их вероятность верного детектирования ( $w_b$ ) и вычислить взвешенную сумму площадей признаков огня в кадре ( $Z$ ) по формулам 1 и 2.

$$Z_{\text{пламени}} = \sum_{b \in \{\text{регионы с пламенем}\}} w_b \cdot \sum_{(x,y) \in b} 1 \quad (1)$$

$$Z_{\text{дыма}} = \sum_{b \in \{\text{регионы с дымом}\}} w_b \cdot \sum_{(x,y) \in b} 1 \quad (2)$$

По данным взвешенным суммам  $Z$  можно получить средние значения за периоды  $T$ , чтобы понять динамику распространения огня: увеличение или уменьшение его распространения. Данный тренд сравнивается с анализом выходов RNN-модуля, и в итоге принимается окончательное

решение. Использование анализа комбинации двух выходов позволяет увеличить точность модели.

Использование исключительно сети Faster R-CNN дает показатель mAP, равный 88%. Если добавить к ней LSTM сеть, то точность модели составляет 92%. Использование третьего модуля с анализом трендов в течении 10 секунд дает повышение точности до 97%. Следует отметить, что с ростом времени анализа тренда растет точность модели (вплоть до 100%) при анализе видеоряда в течении 2 минут.

**Список использованных источников:**

1. Object Detection: Speed and Accuracy Comparison (Faster R-CNN, R-FCN, SSD, FPN, RetinaNet and YOLO) [Электронный ресурс]. – Электронные данные. – Режим доступа: [https://medium.com/@jonathan\\_hui/object-detection-speed-and-accuracy-comparison-faster-r-cnn-r-fcn-ssd-and-yolo-5425656ae359](https://medium.com/@jonathan_hui/object-detection-speed-and-accuracy-comparison-faster-r-cnn-r-fcn-ssd-and-yolo-5425656ae359) – Дата доступа: 25.03.20.
2. A Video-Based Fire Detection Using Deep Learning Models / B. Kim and J. Lee // Applied Sciences, 2019. – P. 8.

## МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ КОНТРОЛЯ ПИТАНИЯ

*Долматович А.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Калугина М.А. – канд. физ.-мат. наук, доцент*

Мобильное приложение в современном мире является помощником во многих сферах жизни. Большой популярностью пользуются программные средства, помогающие контролировать свое здоровье и питание. Вместе с ростом популярности и количества подобного рода приложений растет и спрос к их функциональным особенностям. В докладе анализируются функциональные особенности одного из полученных решений в этой области.

Представленное к рассмотрению мобильное приложение является по своей сути анализатором состава продуктов питания на наличие пищевых добавок. Используемая для анализа информация поступает путем оптического распознавания текста с этикетки исследуемого продукта, полученной путем сканирования с помощью камеры мобильного устройства [1].

В качестве основных функциональных особенностей программы можно выделить следующие ее свойства:

- возможность получать список пищевых добавок, найденных на этикетке продукта питания;
- работоспособность в офлайн режиме;
- описание влияния на организм;
- получение справочной информации о добавках;
- история просмотров и поиска.

Пищевые добавки - вещества, добавляемые в продукты питания в процессе производства, упаковки, транспортировки или хранения для придания им желаемых свойств. К ним можно отнести, например, определенный аромат (ароматизаторы), цвет (красители), длительность хранения (консерванты), вкус, консистенция и т. д. [2]. В Европе пищевые добавки маркируются цифровым кодом с идущей впереди буквой «Е».

Оптическое распознавание символов — перевод изображений рукописного, машинописного или печатного текста в текстовые данные, использующиеся для представления символов в компьютере. Оптическое распознавание символов позволяет редактировать текст, осуществлять поиск слов или фраз, хранить его в более компактной форме, анализировать информацию, а также применять к тексту электронный перевод, форматирование или иное преобразование в нужную форму.

Для осуществления задуманной идеи были выбраны современные и оптимальные технологии, такие как язык программирования Swift [3], способ хранения данных (база данных Realm), способ обработки изображений (GPUImage) и способ оптического распознавания текста (Tesseract).

GPUImage загружает и модифицирует изображение, полученное с камеры мобильного устройства, пропуская его через цепочку фильтров и выдавая результат в виде объекта класса UIImage. Это изображение отдается на обработку в Tesseract.

Tesseract является свободной библиотекой. Для платформы iOS существует собственная обертка, которая совместима с языками Swift и Objective-c. Библиотека предоставляет возможность распознать около 70 наиболее популярных шрифтов и более 100 языков.

Для хранения истории и информации о пищевых добавках была смоделирована база данных [4], которая служит основой приложения. Структура базы данных изображена на рис. 1.

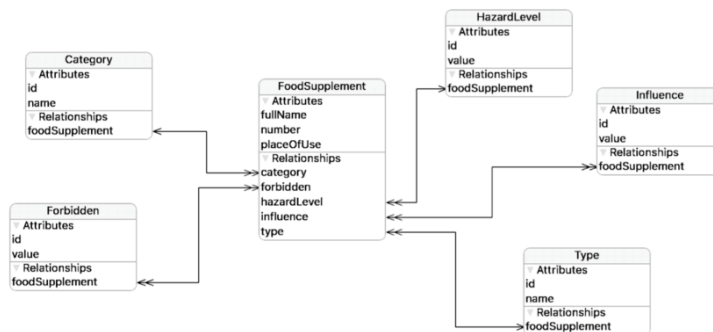


Рисунок 1 - Модель базы данных программного средства

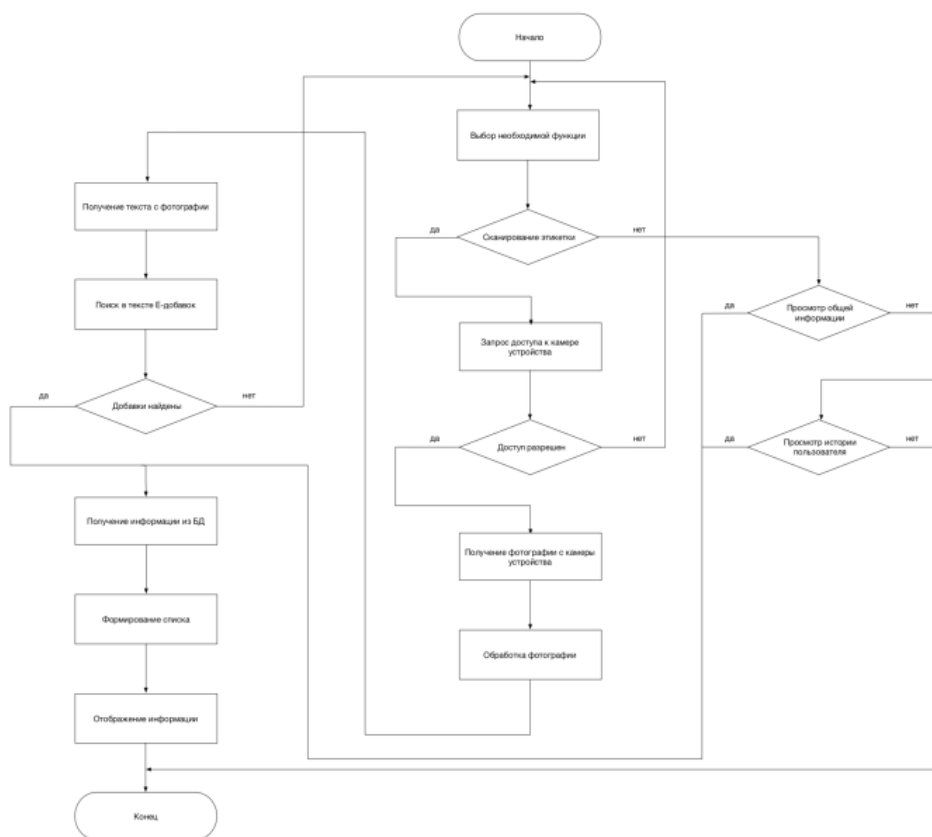


Рисунок 2 — Алгоритм работы программы

Для реализации задуманного функционала было произведено проектирование программного средства, разработана программная архитектура и алгоритмы основных функций программного средства, в том числе общий алгоритм приложения, блок-схема которого изображена на рис. 2.

Для улучшения приложения в дальнейшем необходимо доработать способы обработки фотографий, что позволит еще больше улучшить качество распознавания, а так же дизайн самого приложения. Возможно, расширение функционала, что сделает приложение более универсальным и конкурентно-способным на рынке.

**Список использованных источников:**

1. Мобильные приложения [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://appliftonews.ru/>. – Дата доступа: 13.02.20
2. Пищевые добавки [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://toitumine.ee/>. – Дата доступа: 15.01.20
3. Усов, В.А. Swift. Основы разработки приложений под iOS / В.А. Усов - Санкт-Петербург: Питер, 2015 - 320с
4. Realm [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://realm.io>. – Дата доступа: 03.01.20.

## СИСТЕМА РЕКОМЕНДАЦИИ ВИДЕОКОНТЕНТА

Калоша А.Л.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Стержанов М.В. – к.т.н., доцент

Аннотация. Цель данной работы заключается в создании системы для рекомендации видеоматериалов. В качестве видеоматериалов были выбраны фильмы и сериалы с сайта kinopoisk. В данной системе используется нейронная сеть, которая обучена на наборе метрик. В качестве набора метрик используется название, описание, список актеров, год выпуска. Обучение нейронной сети производилось на 10 000 фильмах. В результате обучения нейронная сеть способна рекомендовать довольно интересные подборки фильмов и сериалов.

Идея рекомендательных систем видеоматериалов не нова в современном мире. Как правило, в данных системах производится поиск людей с похожими интересами. Однако это не всегда работает, так как многим нравятся смежные жанры. Для решения данной проблемы отлично подойдет система на основе нейронной сети.

Данная система обучена для рекомендации фильмов, однако она может быть обучена для рекомендации видеоматериалов с любого веб-ресурса. Сбор данных с веб-сайтов возможен при помощи веб-краулеров. В данном исследовании для этих целей использовался Selenium WebDriver.

Система рекомендации обучается на основе оценок уже просмотренных видеоматериалов, а также по метрикам самих видео.

Для обучения нейронной сети была выбрана библиотека TensorFlow как один из лучших инструментов машинного обучения. TensorFlow — это библиотека программного обеспечения с открытым исходным кодом для численного расчета с использованием графиков потока данных [1].

Для работы системы был создан пользовательский интерфейс, на котором можно выставлять оценки и просматривать рекомендованные системой видео. Список фильмов отсортирован по рекомендациям нейронной сети, то есть фильмы на первых страницах предпочтительнее фильмам на последних. Пример пользовательского интерфейса приведен на рисунке 1.

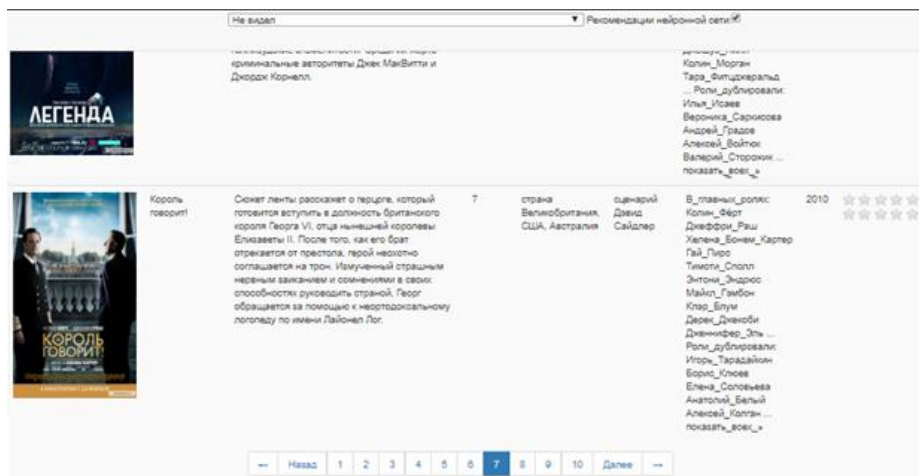


Рисунок 1 – Пример пользовательского интерфейса

Точность рекомендаций системы хорошая. Многие интересные фильмы, не оцененные в данном сервисе, были порекомендованы, а также много новых. Некоторые категории фильмов можно наблюдать в конце списка. К таким, например, относятся аниме или старые фильмы в плохом качестве. Это также свидетельствует о высокой точности обучения системы, так как при обучении фильмам из данных категорий выставлялись минимальные оценки.

### Список использованных источников:

1. Library for numerical computation using data flow graphs [Электронный ресурс]. – Режим доступа: <https://www.tensorflow.org/>. – Дата доступа: 15.04.2020г.



## СВЕРТОЧНАЯ НЕЙРОННАЯ СЕТЬ

Клапатов И.А., Чибисов И.В., Климович М.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Стержанов М.В. – кандидат технических наук, доцент

Сверточные нейронные сети - это такие структуры, которые предназначены для распознавания образов, а именно решающий задачи классификации, задача нахождения позиции объекта, задача переноса стиля, а также векторизации.

Данные нейронные сети используют операцию свертки для работы с данными. Свертка подразумевает работу с абстракцией данных, при этом данная структура сама определяет, с абстракциями какого порядка она будет работать. Последовательная послонная свертка позволяет использовать одни части нейронных сетей решающих одну задачу для решения совершенно другой. Однако, такие слои, несмотря на свою эффективность, остаются слишком сложными для понимания и структуризации.

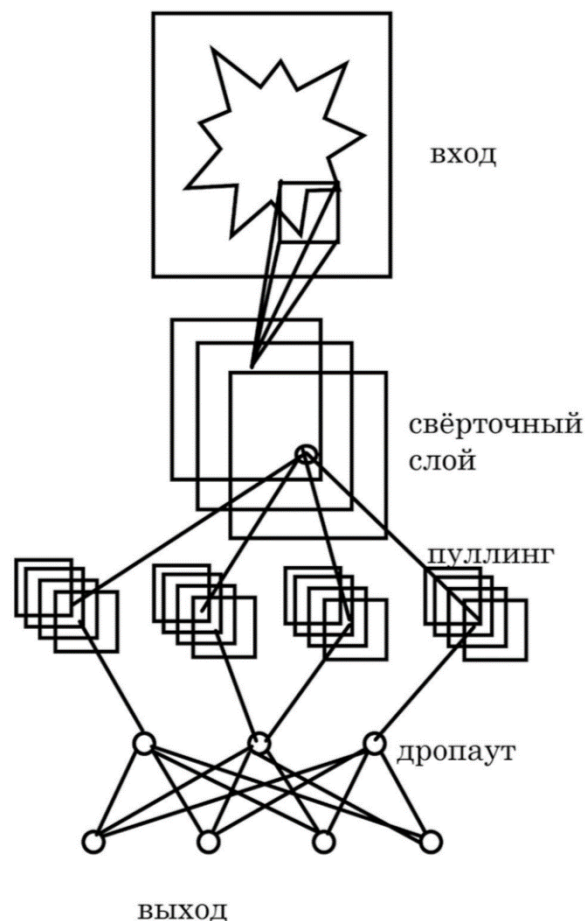


рисунок 1 архитектура CNN

Основным компонентом сети как раз является сверточный слой. Данный слой пофрагментно обрабатывает предыдущий, поэлементно суммируя результаты из каждого фрагмента, а также применяя фильтр.

После каждого слоя свертки следует слой активации, который скалярно складывает элементы сверточного слоя и передает результат далее. Слой активации представляет собой любую нелинейную функцию, например ReLU. Также логически сверточный слой и слой активации группируют в один.

Далее идёт слой уплотнения (пуллинга). Обычно данный слой имеет меньшую размерность и в неё пофрагментно, используя функцию максимума загружают значения. Данный слой вычленяет признаки из предыдущего слоя. К тому же, такая фильтрация признаков позволяет предотвратить переобучение. Кроме функции максимуму, ещё можно использовать и другие функции. Также

применяется функция L2-нормирования.

После последовательности слоёв свертки и пуллинга идёт полносвязную нейронную сеть. Она может также состоять из нескольких слоёв.

Обучать данную сеть надо с учителем, т.е. на размеченных данных и использовать метод обратного распространения ошибки. Для увеличения эффективности сверточной нейронной сети используют слои дропаута. Они позволяют удалять одиночные нейроны из сети, что позволит предотвратить переобучение.

Главным преимуществом данной архитектуры нейронной сети является лучшая производительность для задачи классификации изображений. По сравнению с полносвязной нейронной сетью в сверточной сети снижено количество настраиваемых весов, что позволяет использовать обобщение фрагментов вместо попиксельной обработки изображения. Также данная сеть обладает возможностью параллельной работы. Так как для вычисления слоя можно параллельно высчитывать значения для разных фрагментов изображения. Этот метод устойчив к аугментации, т.е. к поворотам и изменениям изображения - сверточная сеть будет продолжать выдавать правильный результат.

К недостаткам сети можно отнести большое количество параметров для настройки. Все эти параметры существенно влияют на результат, но выбираются исследователями эмпирически. Существует несколько выверенных и прекрасно работающих конфигураций сетей, но не хватает рекомендаций, по которым нужно строить сеть для новой задачи.

Данная нейронная сеть является одним из лучших алгоритмов по распознаванию и классификации изображений.

**Список использованных источников:**

1. DeepLearning 0.1. LISA Lab [Electronic resource]. - Mode of access: <http://deeplearning.net/tutorial/lenet.html> - Date of access: 10.04.2020.
2. Matusugu, Masakazu; Katsuhiko Mori; Yusuke Mitari; Yuji Kaneda. Subject independent facial expression recognition with robust face detection using a convolutional neural network. [Electronic resource]. - Mode of access: [http://www.iro.umontreal.ca/~pift6080/H09/documents/papers/sparse/matsugo\\_et\\_al\\_face\\_expression\\_conv\\_nnet.pdf](http://www.iro.umontreal.ca/~pift6080/H09/documents/papers/sparse/matsugo_et_al_face_expression_conv_nnet.pdf) - Date of access: 13.04.2020.

## ПРОГНОЗИРОВАНИЕ ПОЛОЖЕНИЯ СПУТНИКОВ

*Клебанов Д. А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Калугина М. А. — канд. физ.-мат. наук, доцент*

В докладе описывается суть задачи прогнозирования положения спутников, рассматривается алгоритм её решения и анализируются полученные результаты при различных наборах входных данных.

Прогнозирование положения спутников – одна из важнейших задач астрономии. Информация о точном положении спутников на орбите может быть использована для маневрирования, расчёта траектории полёта и избегания чрезвычайно опасных столкновений. Любое аварийное сближение может привести не только к разрушению спутников, но и к увеличению объема космического мусора. Например, столкновение Иридиум-Космос в 2009 году увеличило количество космического мусора примерно на 13%. Дальнейшие столкновения могут привести к возникновению синдрома Кесслера, при котором плотность объектов на низкой околоземной орбите резко возрастает и делает недоступным использование космического пространства для многих поколений. Кроме того, более точный прогноз положения спутника поможет рассчитать более эффективные маневры для экономии ракетного топлива и продления срока службы спутника на орбите.

Множество подходов к предсказанию орбиты основаны на физических моделях. На их точность влияют, например, следующие факторы: состояние космического объекта в начале расчета траектории, информация об окружающей среде, такой, как сила тяжести Земли, атмосферное сопротивление и солнечная радиация, а также информация о планируемых маневрах. Однако наши возможности наблюдения и фиксирования показателей окружающей среды сильно ограничены. Зачастую невозможно получить о ней достаточно точную информацию или получение такой информации потребует использования дорогостоящих технологий. Поэтому ошибки предсказаний, основанные только на физических моделях, могут быть слишком большими, чтобы их можно было использовать.

Методы машинного обучения предоставляют другие возможности моделирования и

прогнозирования по сравнению с физическим подходом. Прогноз может быть сделан без явного моделирования космических объектов, маневров космического корабля и космической среды. Вместо этого модели машинного обучения обучаются на основе большого количества наблюдаемых в прошлом данных для прогнозирования будущих событий.

В частности, был реализован алгоритм, прогнозирующий положение спутников в течение следующего месяца [1] на основе данных о положении в прошлом месяце и данных SGP4-симулятора (упрощенная модель возмущений). Положение каждого спутника, а также каждая из трех его координат, рассматривались независимо от других. Анализ исходных данных показал, что для некоторых координат определенных спутников реальное положение отличалось от полученного SGP4-симулятором лишь смещением, вызванным изменением периода вращения. Для таких координат высчитывался новый период, а затем математически рассчитывалось положение спутника в следующем месяце. Для координат, которые отличались от полученных симулятором не только изменением периода, была обучена модель линейной регрессии [2] на данных по этой координате реального положения спутника за прошлый месяц. Линейная регрессия качественно описывает зависимость координат, имеющих период и определенное смещение, от времени. Для предотвращения переобучения моделей была использована L2 регуляризация, которая штрафует модель за большие коэффициенты.

Для реализации данной модели был использован язык программирования Python с множеством библиотек для анализа и обработки данных, в том числе с библиотекой машинного обучения scikit-learn. Данный выбор обусловлен наличием необходимого набора инструментов для анализа и реализации алгоритмов машинного обучения.

В качестве метрики качества алгоритма была использована SMAPE (симметрическая средняя абсолютная процентная ошибка) по каждой из трех координат спутника, а затем взято среднее из значений. Полученная ошибка составляет 10%, что является хорошим показателем для столь длительного периода прогнозирования.

Таким образом, в результате проведенной работы, был реализован алгоритм прогнозирования положения спутников, имеющий высокую скорость работы и низкую ошибку.

**Список использованных источников:**

1. Improving Orbit Prediction Accuracy through Supervised Machine Learning / Hao Peng, Xiaoli Bai // arXiv:1801.04856 [astro-ph.EP], 2018.
2. Машинное обучение / Бринк Хенрик, Ричардс Джозеф, Феверолф Марк, 2017

## **ОБЪЕКТ ДЕТЕКТИОН ДЛЯ АВТОМАТИЗАЦИИ ОБРАБОТКИ ДОКУМЕНТОВ**

*Козак А. В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Теслюк В. Н. – кандидат физ.-мат. наук, доцент*

В данной работе были исследованы и проанализированы различные методы распознавания объектов в условиях высокоструктурированных данных (на примере изображений документов). Также было проведено исследование потенциальных проблем, которые могут препятствовать качественной обработке и извлечению интересующей нас информации, а также их решения.

Компьютерное зрение – это область, которая в последние годы становится все более и более популярной (особенно начиная с развития CNN). И в последнее время центральное место заняли задачи с извлечением полезной информации из документов (изображений документов). Еще одной неотъемлемой частью компьютерного зрения является обнаружение (нахождение) объектов (Object Detection), которая играет важную роль для создания алгоритмов автоматической обработки документов. Обнаружение объектов помогает в оценке скелета объекта, обнаружении структуры документа, наблюдении и т. д. Разница между алгоритмами обнаружения объектов и алгоритмами классификации заключается в том, что в алгоритмах обнаружения мы пытаемся локализовать интересующий объект на изображении. Кроме того, мы можем находить не обязательно только один объект, а находить множество различных объектов. В задаче классификации мы можем только ответить на вопрос: есть ли на заданном изображении интересующий нас объект или нет, что накладывает ограничения на результат работы алгоритма.

Основная причина, по которой мы не можем использовать в решении этой задачи, построение стандартной сверточной сети с последующим полностью связанным слоем, заключается в том, что длина выходного слоя является переменной, а не постоянной, это связано с тем, что число вхождений

интересующих объектов не является фиксированным. Наивный подход к решению этой проблемы состоял бы в том, чтобы взять различные области интереса из изображения и использовать CNN для классификации присутствия объекта в этой области. Проблема такого подхода заключается в том, что интересующие объекты могут иметь различное пространственное расположение внутри изображения и различное соотношение сторон. Следовательно, вам придется выбрать огромное количество областей, а это может быть вычислительно взорвано. Для этого и были разработаны алгоритмы, такие как R-CNN, YOLO и другие, чтобы находить объекты быстро и эффективно.

Стандартная задача обнаружения объектов (т.е. локализация объекта и определение его категории) определяется следующим образом. По изображению целью обнаружения (нахождения, локализации) объекта состоит в том, чтобы определить, существуют ли экземпляры объектов из предопределенных категорий на изображении или нет и, если есть, вернуть пространственное расположение и размер каждого из найденных экземпляров. Большой акцент в задаче локализации делается на обнаружении широкого диапазона естественных категорий, в отличие от обнаружения специфических категорий объектов, где более узкая предварительно определенная категория нашего интереса может присутствовать (фактически одна категория содержит объект другой категории). Хотя тысячи объектов представляют визуальный мир, в котором мы живем, в настоящее время исследовательское сообщество в первую очередь заинтересовано в локализации высокоструктурированных объектов (например, автомобили, лица, велосипеды и самолеты) и сборных объектов (например, людей, коров и лошадей), нежели неструктурированные (например, небо, трава и облака).

В идеале задача обнаружения объектов заключается в разработке универсального алгоритма, который позволяет достичь двух конкурирующих целей: качество / точность и высокая эффективность. Высококачественное обнаружение объектов должно точно определять местоположение и распознавать объекты на изображениях (или видеокдрах), так чтобы можно было находить объекты, различных классов несмотря на большое разнообразие объектов (то есть иметь высокую степень различимости), и локализовать и узнавать экземпляры объектов одной и той же категории, учитывая внутрикласовые вариации (то есть обладать высокой надежностью). Высокая эффективность требует выполнения задачи обнаружения в режиме реального времени с приемлемыми требованиями к потребляемым ресурсам.

Задача автоматической обработки документов главным образом включает в себя нахождение областей интереса. А уже потом в извлечение информации (перевод в цифровой, обрабатываем и анализируем) из областей интереса. Задача нахождения решается как правило задачей локализации объектов на изображении, а для извлечения информации (как правило представленной в буквенно-цифровом или текстовом виде) используются методы OCR, которые также используют алгоритмы распознавания объектов. Стоит отметить, что изображения документов (и, вообще говоря, документы в целом) являются высокоструктурированными данными. Таким образом, к таким данным необходимо применять соответствующие алгоритмы обработки-детекции. Последние время исследования показывают, что с задачей распознавания на высокоструктурированных изображениях справляются лучше всего алгоритмы локализации объектов семейства R-CNN, а именно R-CNN, Fast R-CNN, Faster R-CNN.

Алгоритм R-CNN (сокращено от Region-based Convolutional Neural Networks) был предложен в 2014 году Ross Girshick. Предложенный им алгоритм состоит из трех основных этапов. На первом этапе генерируются потенциальные классонезависимые области. Эти области представляют собой потенциальные распознанные объекты, то есть потенциальные предсказания детектора или области, содержащие необходимые объекты. Второй этап представляет собой большую глубокую сверточную нейронную сеть, которая генерирует фиксированной длины вектор признаков для каждой потенциальной области. И, наконец, третий этап – множество линейных SVM (для каждого класса свой классификатор), которые позволяют определить, что за объект находится в потенциальной области. Если внимательно изучить этапы обучения R-CNN, мы можем легко обнаружить, что обучение модели R-CNN является дорогостоящим и медленным, так как все три этапа являются независимыми и требуют трудоемкой работы.

Чтобы ускорить R-CNN, Girshick усовершенствовал процедуру обучения, объединив три независимых модели в одну совместно обучающую структуру и ускорив общие результаты вычисления предсказаний, назвав полученный алгоритм Fast R-CNN. Вместо того, чтобы извлекать векторы признаков с помощью CNN независимо для каждой потенциальной области, полученной с помощью выборочного поиска, эта модель объединяет их в один прямой проход CNN по всему изображению, а уже потенциальные области совместно используют полученную после прохода матрицу признаков. Затем та же матрица признаков разветвляется для использования каждой областью для применения классификатора объекта и регрессора ограничивающего прямоугольника. Однако, выборочный поиск очень ресурсоемкий алгоритм.

Интуитивное ускоренное решение (т.е. улучшение Fast R-CNN) заключается в интеграции алгоритма генерации потенциальных регионов в саму модель CNN. Faster R-CNN делает именно это:

создает единую унифицированную модель, состоящую из RPN (сети предложения региона) и Fast R-CNN с общими сверточными слоями признаков. Фактически Faster R-CNN состоит из двух частей: RPN и Fast R-CNN. Фактически данный алгоритм использует недавно популярную терминологию нейронных сетей с механизмами «внимания», модуль RPN сообщает модулю Fast R-CNN, где стоит искать объект.

В своей работе я проанализировал работу каждого из алгоритмов семейства R-CNN на предмет качественного обнаружения интересующих областей в документах, то есть с областями данных необходимых для извлечения. Полученные результаты показали, что наиболее эффективный и качественный алгоритм для работы с высокоструктурированными данными является алгоритм Faster R-CNN. Это объясняется тем, что только в этой архитектуре используется обучаемый модуль по извлечению потенциальных областей, который потом используются для детектирования. Алгоритмы R-CNN и Fast R-CNN же используют выборочный поиск, который плохо показывает себя в работе со структурированными изображениями, так как является детерминированным алгоритмом, основанным на необучаемом алгоритме попиксельной иерархической сегментации.

Полученные результаты и реализованные методы были использованы в разработке ПО автоматической обработки и оценивания контрольных и / или проверочных работ студентов и / или учащихся в средних и / или высших учебных заведениях и исследовании применение методов машинного обучения в системе образования.

**Список использованных источников:**

1. Rich feature hierarchies for accurate object detection and semantic segmentation [Электронный ресурс]. — Режим доступа: <https://arxiv.org/pdf/1311.2524.pdf>
2. Fast R-CNN [Электронный ресурс]. — Режим доступа: <https://arxiv.org/pdf/1504.08083.pdf>
3. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks [Электронный ресурс]. — Режим доступа: <https://arxiv.org/pdf/1506.01497.pdf>

## **ТРАНЗАКЦИИ БЛОКЧЕЙН СИСТЕМ: СОДЕРЖАНИЕ И ЖИЗНЕННЫЙ ЦИКЛ**

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Кордаш В. Г., Бирич С.С.*

*Скобцов В.Ю. – к.т.н., доцент*

Работа посвящена вопросам исследования транзакций распределенных систем с целью осознания необходимости всех процессов, поддерживающих состояние и подлинность распределенных систем.

Распределенная система — это совокупность независимых устройств, связанных между собой определенным программным комплексом, позволяющим пользователю рассматривать их как единую систему [1]. Блокчейн – один из способов организации такой системы, декларирующий механизм сохранения и обмена информацией. Существуют два типа подобных систем: открытые и закрытые.

Каждая блокчейн система согласует действия своих компонентов и узлов используя определенные протоколы. В зависимости от концептуальных особенностей узлы могут образовывать иерархические зависимости, или быть равными. Несмотря на то, что изначально открытые блокчейны использовали только равные типы узлов, на данный момент данная опция доступна каждой категории.

Иерархическая структура необходима для распределения нагрузки, а также для разделения функций и полномочий узлов, что может в определенных условиях положительно сказаться на характеристиках системы, таких как надежность, отказоустойчивость и скорость обработки вносимой информации.

Простейшим завершённым элементом данных является транзакция. Традиционной схемой ассамблирования транзакций является блок. Однако в нем могут содержаться совершенно независимые транзакции. Следовательно операции с блоками не являются атомарными. Транзакция может содержать множество полей. Необходимым содержанием является адрес. В зависимости от свойств и назначения системы данный адрес может ссылаться на самые различные понятия: кошелек, контракт, чейнкод, текст и другие. Помимо адреса правильная транзакция содержит ещё какую-либо полезную нагрузку. Данная полезная нагрузка и определяет набор дальнейших действий. Но без привязки к какому-либо адресу она не имеет смысла, ровно как и адрес в отрыве от контекста невозможно использовать. Таким образом, блокчейн следует рассматривать как набор структурированных транзакций.

Для того, чтобы понять роль транзакций будут продемонстрированы жизненные циклы

транзакции открытой системы и закрытой (Hyperledger). Однако следует помнить, что в зависимости от реализации внутренних механизмов жизненные циклы других подобных систем будут отличаться, а также чем более проста система, тем больше пунктов данного процесса в ней может быть вырожденно до простейших операций.

Ниже видны несколько жизненных циклов транзакций публичного Блокчейна. Общая последовательность шагов:

а) Кто-то запрашивает транзакцию с помощью кошелька.

б) Транзакция отправляется (широковещательно) всем участвующим компьютерам в определенной сети цепочки блоков.

в) Каждый компьютер в сети проверяет (валидирует) транзакцию на соответствие некоторым правилам валидации, которые установлены создателями определенной сети цепочки блоков.

г) Проверенные транзакции хранятся в блоке и блокируются (хэшем).

д) Этот блок становится частью цепочки блоков, когда другие компьютеры в сети проверяют правильность хэша блока.

е) Теперь транзакция является частью блокчейна и никак не может быть изменена [2].

Далее находится процесс сохранения транзакции в сети Hyperledger:

а) Пользователь инициирует предложение транзакции, а также подписывает ее.

б) Подтверждающие пиры, верифицируют подпись и исполняют транзакцию. В зависимости от результатов исполнения они формируют ответы на предложения и изменения в состояние.

в) Ответы на предложения подлежат проверке там проверяется их идентичность.

г) Пользователь собирает транзакцию из подтверждений и отправляет управляющему порядком транзакций узлу.

д) Транзакция проходит валидацию и фиксируется.

е) Обновляется содержание журнала и состояние. Журнал хранит запись о транзакции вне зависимости, была она одобрена или нет [3].

Рассмотрев подробно жизненные циклы открытой и закрытой системы, можно увидеть значительную диверсификацию узлов в закрытой сети. Каждый узел отвечает за свой узкий объем работы по поддержанию гомеостаза системы. Это позволяет упростить вычисления и увеличить поток транзакций, так как они в процессе занесения в журнал будут проходить параллельные этапы. Также такая многоуровневость позволяет уменьшить риск возникновения ошибок, как случайных, так и привнесенных намеренно. На каждом этапе происходит проверка подлинности предыдущего и на состояние системы влияет только верная транзакция. Также важным моментом является тот факт, что каждая из попыток обновления системы посредством внесения транзакций фиксируется, что позволяет производить неподделываемую аналитику, так как все изменения зафиксированы без возможности удаления.

Жизненный цикл публичного блокчейна не обладает столькими механизмами защиты, в них нет необходимости. Так как масштабы узлов велики, данная система не смогла бы поддерживать столь большой набор сообщений, необходимый для учета транзакции.

**Список использованных источников:**

1. Э. Таненбаум, М. Ван Стеен. Распределенные системы. Принципы и парадигмы // СПб.: Питер, 2003. 877 с.
2. Pulling the Blockchain apart.. The transaction life-cycle [Электронный ресурс]. — Режим доступа : <https://medium.com/ignation/pulling-the-blockchain-apart-the-transaction-life-cycle-7a1465d75fa3> Дата доступа : 01.03.2020.
3. Hyperledger: transaction flow [Электронный ресурс]. — Режим доступа : <https://www.ibm.com/blockchain/hyperledger> — Дата доступа : 01.03.2020.

## **АНАЛИЗ АППАРАТУРНЫХ ЗАТРАТ НА РЕАЛИЗАЦИЮ ЦИФРОВОГО УСТРОЙСТВА ВСЕВОЗМОЖНЫХ ПЕРЕСТАНОВОК**

*Кохновский С.И.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Иванюк А.А. – д.т.н., профессор*

Целью работы является анализ аппаратных затрат при размещении цифрового устройства всевозможных перестановок на кристалле семейства Artix 7.

В современном мире как никогда актуальна проблема защиты авторских прав. Все объекты интеллектуальной собственности могут подвергаться атакам злоумышленников в совершенно непредсказуемое время. Для обеспечения большей трудоёмкости доступа к защищенной информации применяются различные методы, в том числе запутывающие устройства, однозначно отображающие входные сигналы на выходные в зависимости от подаваемого ключа. Далее будет

рассмотрено одно из таких устройств.

На входы комбинационной схемы подаются  $N$  бит входных сигналов и  $M$  бит ключа. Стоит заметить, что  $M = \frac{N}{2} * (N - 1)$ . На выходы подаются  $N$  бит – запутанная последовательность. Пусть рассматривается устройство размерностью  $N = 2$ . Вычислив  $M$ , получим 1. Таким образом, соответствие входных и выходных сигналов одинакового порядкового номера будет прямым, если ключ был равен 0, и обратным в противном случае. Обобщение работы устройства на размерность любого  $N$  представлено в работе [1].

Следует рассмотреть характеристики и оценить аппаратные затраты на реализацию цифрового устройства, которое может быть расположено на кристалле серии Artix-7 компании Xilinx. Кристаллы программируемой логики серии Artix-7 относятся к ПЛИС, эффективно используемым в настоящее время [2].

В процессе синтеза комбинационного устройства для  $N = 18$  исчерпывается количество портов ввода и вывода, что обусловлено аппаратными характеристиками кристалла. На кристалле XC7A200T, который использовался для синтеза и моделирования устройства, имеется лишь 500 портов. Решением проблемы дефицита портов является механизм периферийного сканирования: последовательно передавая сигналы, которые поступают на вход, и сигналы ключа, есть возможность добиться сокращения количества используемых входов и выходов на кристалле до константы – 5, однако появляется необходимость в использовании сигналов синхронизации *clk* и разрешения *en*. После внедрения вышеописанных измерений комбинационная схема становится последовательной цифровой схемой с некоторой комбинационной частью (рисунок 1). Очевидно, что при увеличении размерности комбинационной части, ни логика, ни аппаратные затраты на реализацию последовательной части не возрастают, будет изменяться лишь количество тактов между повторной установкой сигнала *en*.

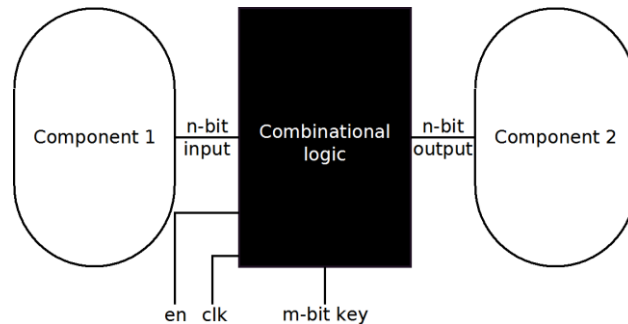


Рисунок 1. Схема последовательного цифрового устройства

Аппаратные ресурсы ПЛИС ограничены количеством срезов (slices), которые располагаются на кристалле. Срезы, в свою очередь, состоят из справочных таблиц (look-up tables) и триггеров (flip flops). Количество справочных таблиц и триггеров, которые Xilinx определяет для создания одного среза, различается в зависимости от семейства кристалла. Кристалл XC7A200T имеет 134600 срезов. Измерения количества использованных срезов для устройств с количеством входов 4, 8, 16, 32, 64 отмечены точками (рисунок 2). Методом степенной регрессии получен график ожидаемой зависимости количества использованных срезов, от количества входных сигналов.

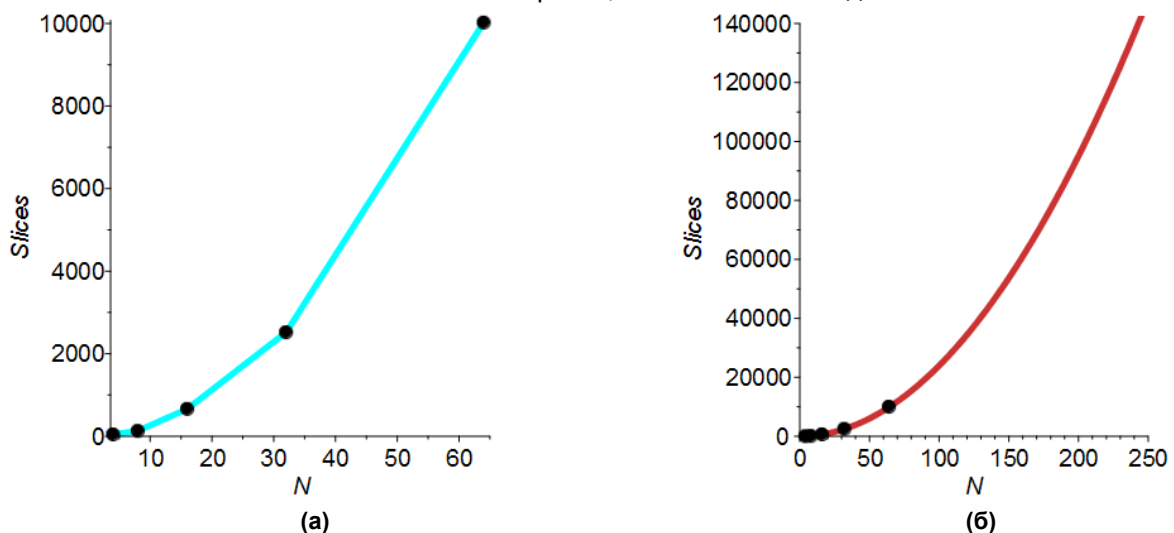


Рисунок 2. Практическая зависимость использованного количества срезов от количества входных сигналов (а), теоретическая зависимость использованного количества срезов от количества входных сигналов (б)

Согласно полученному уравнению регрессии, максимальная ожидаемая размерность устройства, которое можно разместить на кристалле XC7A200T, составляет  $N = 238$ . Также интерес представляет такой параметр, как частота [3]. Опытным путём было замечено, что с возрастанием размерности устройства, частота снижается (рисунок 3).

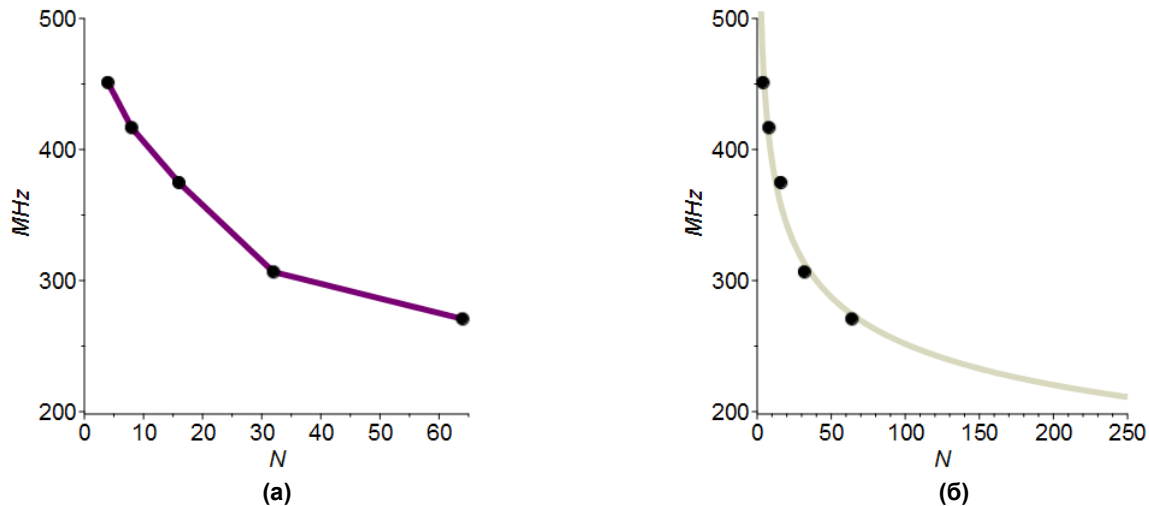


Рисунок 3. Практическая зависимость частоты от количества входных сигналов (а), теоритическая зависимость частоты от количества входных сигналов (б)

Результатом данной работы является приведённая оценка аппаратных затрат для изготовления цифрового устройства всевозможных перестановок. Так как кристаллы серии Artix-7 обладают относительно невысокой ценой, уменьшенным энергопотреблением по сравнению с предыдущими семействами, их использование вполне оправдано для массового производства цифровых устройств. Устройства могут использоваться для усложнения задач обратного проектирования систем как малой, так и относительно большой сложности. Однако следует учитывать, что увеличение количества входных сигналов ведёт не только к потере производительности, но и сильно ограничено сверху. Ограничения можно расширить, используя кристаллы большей ресурсоёмкости, тем не менее это приведёт к росту цены конечного продукта.

**Список использованных источников:**

1. Кохновский, С. И. Методика проектирования комбинационного устройства всевозможных перестановок [Электронный ресурс] / С. И. Кохновский // Компьютерные системы и сети : сб. тезисов докладов. – БГУИР, Минск, 2019. – 287 с. – С. 205-207. – Режим доступа: [https://www.bsuir.by/m/12\\_100229\\_1\\_136895.pdf](https://www.bsuir.by/m/12_100229_1_136895.pdf). – Дата доступа: 22.03.2020.
2. Artix-7 Product Advantage [Electronic resource]. – Mode of access: <https://www.xilinx.com/products/silicon-devices/fpga/artix-7.html>. – Date of access: 22.03.2020.
3. Clock Signals in FPGA Design: Data Path Maximal Clock Rates and the Xilinx PERIOD Timing Constraint [Electronic resource]. – Mode of access: <https://www.allaboutcircuits.com/technical-articles/clock-signal-FPGA-design-clock-rate-xilinx-period-timing-constraint/>. – Date of access: 22.03.2020.

## КРИПТОГРАФИЧЕСКИЕ АЛГОРИТМЫ ШИФРОВ ЗАМЕНЫ

*Круглая А. А., Кулевич А. О.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Стройникова Е. Д. – ст. преп. кафедры информатики*

В работе исследованы существующие алгоритмы шифров замены и методы их реализации. Зашифровывание и расшифровывание с использованием ряда алгоритмов продемонстрировано на примерах. Реализованы учебные программы на языке программирования C# для ознакомления и изучения студентами. Преимуществами программных кодов являются их понятность и доступность.

Современный мир очень сложно представить без интернета и социальных сетей. Основой передачи информации в интернете является шифрование, так, самое простое сообщение «Привет» успевает быстрее, чем за секунду, зашифроваться у одного пользователя и расшифроваться у другого. Изучением шифрования занимается криптография – наука и искусство передачи сообщений в таком виде, чтобы их нельзя было прочитать, не зная специального секретного ключа. В наши дни криптография характеризуется использованием открытых алгоритмов шифрования.

Шифры разделяются на два больших класса: шифры замены и шифры перестановки. Данная



работа посвящена исследованию алгоритмов шифров замены (или подстановки), разновидности которых представлены на рисунке 1.

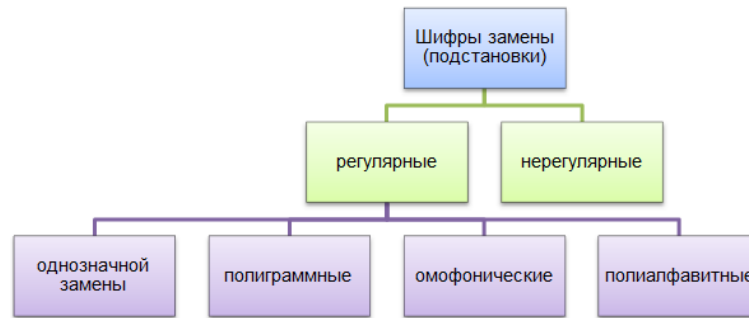


Рисунок 1 – Разновидности шифров подстановки

**Принцип шифрования методом замены.** Каждую букву открытого сообщения зашифровывают, сопоставляя её с каким-то множеством шифрозамен, где первой букве соответствует некоторое множество  $M_1$ , второй –  $M_2$  и т. д. Шифрозамены выбираются таким образом, чтобы любые два множества  $M_i$  и  $M_j$  при  $i \neq j$  не содержали одинаковых элементов, т. е.  $M_i \cap M_j = \emptyset$ .

**1. Регулярные шифры** – шифры, в которых шифрозамены состоят из строго определённого количества символов или в шифрограмме они отделяются друг от друга пробелом, точкой, тире и т. п.

**1. Шифры однозначной замены** (или моноалфавитные, простые подстановочные) характеризуются тем, что количество шифрозамен для каждого символа исходного алфавита равно 1.

Шифр **Атбаш** состоит в замене каждой буквы другой буквой, которая находится в алфавите на таком же расстоянии от конца, что и оригинальная буква от начала. Например, в английском алфавите буква «А» заменяется на «Z», буква «В» – на «Y» и т. д. Таким образом, сообщение «INFORMATICS» зашифруется в виде «RMULINZGRXH». Программа, реализующая алгоритм шифра Атбаш, доступна по ссылке: <https://pastebin.com/vWG8PBR3>.

Шифр **Цезаря**. Во время шифрования каждая буква заменяется другой, отстоящей от неё в циклически записанном алфавите на фиксированное число позиций. Программа доступна по ссылке: <https://pastebin.com/ssjdMDcD>. **ROT13** является современным примером шифра Цезаря. В этом шифре каждый символ английского алфавита циклически сдвигается на 13 позиций (рисунок 2). Если зашифруем сообщение «KSIS», то получим сообщение «XFVF». А если зашифруем «XFVF» (или, что то же самое, расшифруем его), то получим снова «KSIS».

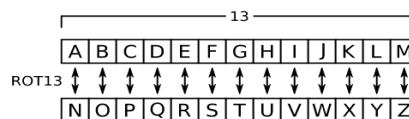


Рисунок 2 – Преобразование букв английского алфавита при помощи ROT13

**2. Полиграммные шифры замены.** В данном виде шифрования одной шифрозамене может соответствовать сразу несколько символов исходного текста.

**Биграммный шифр Порты** – первый известный биграммный шифр, который представляется в виде квадратной таблицы. Вертикально слева и горизонтально сверху находится алфавит, в самих ячейках записываются любые символы. Главное, чтобы содержимое ячеек не повторялось. Сообщение шифруется парами букв исходного текста. Первая буква указывает на строку шифрозамены, а вторая – на столбец. При нечётном количестве букв в исходном сообщении к нему добавляется вспомогательный символ. К примеру, если при помощи таблицы шифрозамен для шифра Порты, приведённой в [1], зашифруем слово «ИН ФО РМ АТ ИК А», то получим «261 603 477 018 258 031». В качестве вспомогательного символа использована буква «Я».

**3. Омофонические шифры** (или однозвучные, многозначной замены) характеризуются тем, что для отдельных символов исходного алфавита количество шифрозамен больше 1 ( $|M_i| \geq 1$  для одного символа). Примером омофонического шифра является **книжный шифр**, в котором каждый элемент открытого текста (каждая буква или слово) заменяется на указатель (например номер страницы, строки и столбца) аналогичного элемента в дополнительном тексте-ключе.

**4. Полиалфавитные шифры** состоят из нескольких шифров однозначной замены и отличаются друг от друга способами выбора варианта алфавита для шифрования одного символа.

**Таблица Трисемуса** – многоалфавитный шифр, основанный на таблице размером  $n \times n$ , где  $n$  – количество символов в алфавите. В первой строке таблицы записываются буквы алфавита в порядке их очерёдности, во второй – та же последовательность букв с циклическим сдвигом на одну позицию

влево, в третьей – с циклическим сдвигом на две позиции влево и т. д. Исходный текст состоит из букв первой строки,  $i$ -я буква сообщения зашифровывается буквой, находящейся в  $i$ -й строке, в том же столбце, что и исходная буква. Если длина текста больше  $n$ , то после использования последней строки вновь возвращаются к первой и т. д. Так, используя таблицу размером  $33 \times 33$  для русского алфавита, сообщение «БГУИР» зашифруется в виде «БДХЛФ».

**Система шифрования Виженера** основана на таблице Трисемуса, отличие от предыдущего метода шифрования заключается в том, что вначале выбирается ключ, который состоит из символов алфавита,  $i$ -я буква исходного сообщения зашифровывается буквой, находящейся в строке, начинающейся с  $i$ -й буквы ключа, в том же столбце, что и исходная буква. Если длина ключа меньше длины сообщения, то ключ используется повторно. Например, зашифровывая сообщение «ИНФОРМАТИКА» с использованием таблицы размером  $33 \times 33$  для русского алфавита и ключа «КСИС», получим шифrogramму «УЯЭАЫЮИДУЫИ». Программа доступна по ссылке: <https://pastebin.com/aDjc79YY>.

**II. Нерегулярные шифры.** В шифрах данного типа шифрозамены состоят из разного количества символов и записываются в шифrogramме подряд (без отделения друг от друга), что значительно затрудняет криптоанализ.

**Совмещённый шифр** (или совмещённая таблица), названный так из-за необычного использования одно- и двухцифровых шифрозамен. Таблица шифрозамен содержит десять столбцов с нумерацией 0, 9, 8, 7, 6, 5, 4, 3, 2, 1. В начальной строке записано слово-ключ, исключая повторяющиеся буквы. В следующие строки вписываются по десять не вошедших в слово-ключ букв в порядке их следования в алфавите. Все строки, кроме начальной, нумеруются по порядку, начиная с 1. Во время шифрования, если буква исходного сообщения совпадает с какой-то буквой слова-ключа, то она заменяется одной цифрой – номером столбца, в противном случае – двумя – номерами строки и столбца. Так, сообщение «ИНФОРМАТИКА» зашифруется «146242028117261487» (см. рисунок 3).

	0	9	8	7	6	5	4	3	2	1
	Д	Е	К	А	Н					
1	Б	В	Г	Ж	З	И	Й	Л	М	
2	О	П	Р	С	Т	У	Ф	Х	Ц	Ч
3	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	-	-

Рисунок 3 – Пример таблицы шифрозамен совмещённого шифра с ключом «ДЕКАН»

Основным приёмом при вскрытии криптограмм шифров замены является анализ частоты встречаемости букв в исходном тексте. Несмотря на вытеснение шифров замены блочными шифрами, одноразовые блокноты, основанные на шифре Вернама [2], продолжают использоваться на государственном уровне в наше время для обеспечения сверхсекретных каналов связи.

Разработанные авторами на языке C# программы, реализующие ряд алгоритмов шифров замены, удобны в использовании благодаря их понятности и доступности. Для большего удобства в кодах программ присутствуют пояснения.

**Список использованных источников:**

1. Анисимов В. В. [Криптографические методы защиты информации](https://sites.google.com/site/anisimovkhv/learning/kripto/lecture/tema4). Шифры замены [Электронный ресурс]. – Режим доступа : <https://sites.google.com/site/anisimovkhv/learning/kripto/lecture/tema4>.
2. Свободная энциклопедия Википедия. Шифр Вернама [Электронный ресурс]. – Режим доступа : [https://ru.wikipedia.org/wiki/Шифр\\_Вернама](https://ru.wikipedia.org/wiki/Шифр_Вернама).

## РЕАЛИЗАЦИЯ КРИПТОГРАФИЧЕСКИХ МЕТОДОВ НА ЯЗЫКЕ C++

*Куделя А. А., Макарич Д. А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Стройникова Е. Д. – ст. преп. кафедры информатики*

В работе исследованы некоторые симметричные и асимметричные криптографические алгоритмы, осуществлена их реализация на языке программирования C++, проведён анализ результатов программ, выявлены оптимальные случаи использования тех или иных криптографических методов.

В настоящее время, с развитием информационных технологий, обеспечение секретности и приватности при коммуникациях между людьми является одной из основных задач. Криптология – наука, исследующая криптографические преобразования, т. е. преобразования, которые обеспечивают защиту информации от несанкционированного доступа. В криптологии различают следующие направления: криптографию и криптоанализ. Криптография – это часть криптологии, связанная с проектированием секретных систем. Криптоанализ – это часть криптологии, посвящённая изучению уязвимости секретных

систем.

В работе были изучены симметричные (с закрытым ключом) и асимметричные (с открытым ключом) криптосистемы. В симметричных криптосистемах для шифрования и дешифрования используется общий секретный ключ. В асимметричных же криптосистемах для шифрования используется открытый ключ, а для дешифрования – закрытый ключ. Симметричное шифрование благодаря своей скорости широко используется для защиты информации во многих современных компьютерных системах. Асимметричное шифрование может применяться в системах зашифровывания и расшифровывания пакетов данных, когда скорость и вычислительная мощность не являются приоритетными.

На примере криптографических алгоритмов XOR, Triple DES (3DES) и RSA в работе проведены анализ и сравнение симметричного, симметричного блочного и асимметричного криптографических алгоритмов. Также выявлены случаи наиболее удобного использования того или иного метода шифрования. В практической части реализованы данные и гибридные криптографические методы на языке программирования C++.

XOR – самый простой пример симметричного алгоритма шифрования. XOR – логическая операция, которая принимает значение «истина», если хотя бы один из аргументов имеет значение «истина». Минусом такого шифрования является то, что, зная часть зашифрованного текста, можно с лёгкостью восстановить ключ и, соответственно, расшифровать весь текст. Поэтому в чистом виде данный метод редко используется на практике, хотя его применяют как часть более сложных алгоритмов шифрования.

На рисунке 1 представлен скриншот работы программы, реализующей зашифровывание и расшифровывание сообщения «Hello world!» с помощью алгоритма XOR. В данной программе ключом является строка (string). В цикле, используя операцию XOR, шифруется каждый символ строки, который соответствует коду ASCII, с помощью ключа. Аналогичным методом дешифруется строка в цикле с помощью XOR.

```

Консоль отладки Microsoft Visual Studio
Enter message:
Hello world!
Enter key:
5
Encrypted message:
JPPYZSBZGYN
Decrypt message?
yes
Hello world!
  
```

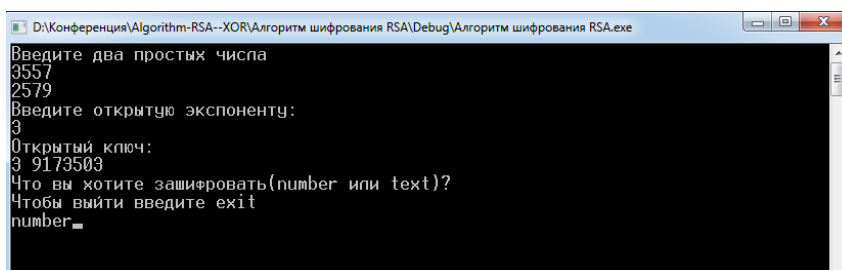
Рисунок 1 – Пример работы программы, реализующей криптографический метод XOR

3DES – симметричный блочный шифр, созданный на основе алгоритма DES. В работе программно реализованы этапы шифрования и дешифрования с использованием сети Фейстеля, изучены криптостойкость и области применения 3DES. Шифр 3DES с тремя различными ключами имеет длину ключа, равную 168 бит, но из-за атак «встреча посередине» эффективная криптостойкость составляет только 112 бит. Для успешной атаки на 3DES потребуется около  $2^{32}$  бит известного открытого текста,  $2^{113}$  шагов,  $2^{90}$  циклов DES-шифрования и  $2^{88}$  бит памяти. На данный момент это непрактично, и, по оценкам НИСТ, алгоритм с выбором трёх различных ключей должен остаться надёжным до 2030-х. В настоящее время известных криптографических атак, применимых на практике, на 3DES не существует. 3DES с тремя ключами реализован во многих приложениях, ориентированных на работу с Интернетом, в том числе в PGP и S/mime. Индустрия электронных платежей использует 3DES и продолжает активно разрабатывать и публиковать стандарты, основанные на нём (например EMV).

Криптографический алгоритм RSA основан на вычислительной сложности факторизации больших чисел. В работе программно реализованы этапы генерации ключей, шифрования и дешифрования алгоритма RSA. Проанализирована скорость работы алгоритма. Поскольку генерация ключей происходит значительно реже операций, реализующих шифрование и дешифрование, задача возведения в степень по модулю представляет основную вычислительную сложность. В программе эта проблема решается с помощью алгоритма быстрого возведения в степень. Атака Винера позволяет узнать расшифровывающую экспоненту  $d$ , поочерёдно подставляя в выражение  $(m^e)^d \equiv m \pmod{n}$  для некоторого случайного  $m$  знаменатели подходящих дробей  $k/d$  из разложения  $e/n$  в непрерывную дробь. RSA используется для защиты программного обеспечения и в схемах цифровой подписи. Из-за низкой скорости шифрования сообщения обычно шифруют с помощью более производительных симметричных алгоритмов со случайным сеансовым ключом, а с помощью RSA шифруют лишь этот ключ, таким образом, реализуется гибридная система.

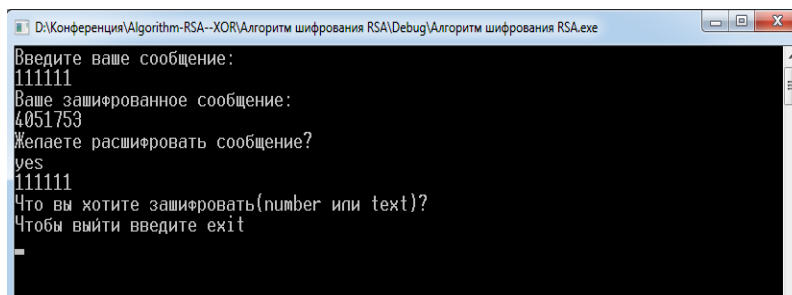
На рисунке 2 представлен скриншот работы программы, реализующей генерацию открытого

ключа (3, 9173503) с помощью алгоритма RSA. На рисунке 3 представлен скриншот работы программы, реализующей зашифрование и расшифрование сообщения «11111» с помощью алгоритма RSA.



```
D:\Конференция\Algorithm-RSA--XOR\Алгоритм шифрования RSA\Debug\Алгоритм шифрования RSA.exe
Введите два простых числа
3557
2579
Введите открытую экспоненту:
3
Открытый ключ:
3 9173503
Что вы хотите зашифровать(number или text)?
Чтобы выйти введите exit
number_
```

Рисунок 2 – Пример работы программы, реализующей генерацию открытого ключа алгоритмом RSA



```
D:\Конференция\Algorithm-RSA--XOR\Алгоритм шифрования RSA\Debug\Алгоритм шифрования RSA.exe
Введите ваше сообщение:
11111
Ваше зашифрованное сообщение:
4051753
Желаете расшифровать сообщение?
yes
11111
Что вы хотите зашифровать(number или text)?
Чтобы выйти введите exit
_
```

Рисунок 3 – Пример работы программы, реализующей зашифрование и расшифрование сообщения алгоритмом RSA

На примерах XOR, Triple DES и RSA рассмотрены особенности программной реализации методов шифрования на языке C++, изучены недостатки и предложены способы их устранения с использованием гибридных алгоритмов. Коды программ доступны по ссылкам:

- 1) <https://github.com/DmitryMakarich/Algorithm-RSA--XOR> – программы RSA, XOR;
- 2) <https://github.com/alexku31/Repp/blob/master/3des.c> – программа 3DES.

**Список использованных источников:**

1. Криптографические методы защиты информации [Электронный ресурс]. – Режим доступа : <https://urok.1sept.ru/%D1%81%D1%82%D0%B0%D1%82%D1%8C%D0%B8/636966/>.
2. Алгоритм RSA [Электронный ресурс]. – Режим доступа : [https://ru.wikipedia.org/wiki/RSA#%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7\\_RSA](https://ru.wikipedia.org/wiki/RSA#%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7_RSA).
3. Алгоритм шифрования XOR [Электронный ресурс]. – Режим доступа : <https://evileg.com/ru/post/271/>.
4. Triple DES [Электронный ресурс]. – Режим доступа : [https://ru.wikipedia.org/wiki/Triple\\_DES#%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D1%81%D1%82%D0%BE%D0%B9%D0%BA%D0%BE%D1%81%D1%82%D1%8C](https://ru.wikipedia.org/wiki/Triple_DES#%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D1%81%D1%82%D0%BE%D0%B9%D0%BA%D0%BE%D1%81%D1%82%D1%8C).
5. DES [Электронный ресурс]. – Режим доступа : [https://ru.wikipedia.org/wiki/DES#%D0%9F%D1%80%D0%B5%D0%BE%D0%B1%D1%80%D0%B0%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F\\_%D1%81%D0%B5%D1%82%D1%8C%D1%8E\\_%D0%A4%D0%B5%D0%B9%D1%81%D1%82%D0%B5%D0%BB%D1%8F](https://ru.wikipedia.org/wiki/DES#%D0%9F%D1%80%D0%B5%D0%BE%D0%B1%D1%80%D0%B0%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D1%81%D0%B5%D1%82%D1%8C%D1%8E_%D0%A4%D0%B5%D0%B9%D1%81%D1%82%D0%B5%D0%BB%D1%8F).
6. Асимметричное шифрование на практике [Электронный ресурс]. – Режим доступа : <https://habr.com/ru/post/449552/>.
7. Системная криптография: использовать FIPS-совместимые алгоритмы для шифрования, хэширования и подписывания [Электронный ресурс]. – Режим доступа : <https://docs.microsoft.com/ru-ru/windows/security/threat-protection/security-policy-settings/system-cryptography-use-fips-compliant-algorithms-for-encryption-hashing-and-signing>.
8. Представлен Sweet32, новый вид атаки на HTTPS и OpenVPN [Электронный ресурс]. – Режим доступа : <https://www.opennet.ru/opennews/art.shtml?num=45023>.

## АВТОМАТИЗИРОВАННАЯ СИСТЕМА КРАУДСОРСИНГА ДАННЫХ

Кудин Н.И., Сабадаш М.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Калугина М.А. – канд. физ.-мат. наук, доцент

В настоящее время системы краудсорсинга активно развиваются, поскольку данный способ сбора информации является эффективным как для рынка Западной Европы, так и для стран СНГ. Данный подход открывает много возможностей для создания уникального сервиса обработки данных, основанных на запросах пользователя.

Современные технологии развиваются ежедневно с высокой скоростью. Одной из проблем, которую приходится решать компаниям в различных сферах, является поиск инновационных идей для дальнейшего развития [1].

Как правило, крупные компании используют высококвалифицированных сотрудников для решения данной проблемы. Но, к сожалению, небольшие компании, как правило, имеют ограниченный бюджет, тем самым не могут себе позволить реализацию инновационных задач в достаточном объеме.

Примером больших компаний, которые находятся в постоянном поиске улучшений условий бизнеса, могут служить British Gas, British Airways, Virgin Trains и другие. Такие крупные «игроки» всегда заботятся о перспективности предлагаемого решения и скорости воплощения идеи в жизнь, так как реализация какого-либо проекта должна выполняться в такие сроки, чтобы идея сохранила свою актуальность [2].

В качестве решения данной проблемы предлагается использование высоконагруженных систем краудсорсинга, которые позволяют в краткие сроки производить анализ профессиональной активности работников компании и выявлять наиболее перспективные идеи для поставленной задачи. Кроме того, данные системы могут организовать отклик сотрудников на проблемные места компании, уменьшить расходы на высококвалифицированные кадры, исключить необходимость постоянного аудита ключевых отделов компании. Такие системы в режиме реального времени дают возможность производить синхронизацию с открытыми источниками данных (например, корпоративные социальные сети). Собрав пакет данных, можно выполнить анализ эффективности работы компании и определить ее слабые звенья [3].

На рисунке 1 представлена схема стандартного процесса краудсорсинга идей.

Примером системы краудсорсинга является Yahoo!Answers [4]. Сервис вопросов и ответов Yahoo!Answers запущен еще в 2005 году. Благодаря сервису любой пользователь может создать вопрос, а все добровольцы из сообщества, у которых есть несколько баллов за предыдущую активность, — попытаться ответить на него. Если хотя бы один из ответов помог решить проблему, пользователь, задававший вопрос, может отметить его как лучший и присвоить ему оценку, которая повышает авторитет пользователя. Большая англоязычная аудитория данного сервиса делает его перспективным для монетизации сервиса.

Предлагаемая нами автоматизированная система краудсорсинга данных предполагает собой абстракцию между источниками данных и непосредственно системой анализа данных. Таким образом, пользователь системы может самостоятельно выбрать необходимые ему источники данных, которые должны быть проанализированы. В свою очередь, система выполняет анализ и отображение данных в виде графиков. Данный способ визуализации помогает пользователям ускорить процесс определения тенденций в источниках данных. Также в системе предусмотрена агрегация данных из разных областей, что поможет определять общий вектор направленности данных.

Особенностью решения является глубокая интеграция сервисов с корпоративными социальными сетями, с помощью которого пользователи смогут решать поставленные задачи. Используя данный подход, целевая аудитория будет расширена многократно.

Алгоритм работы синхронизации является одной из ключевых частей программного продукта. В самом начале работы алгоритма система выбирает все доступные и активные компании в системе. Затем синхронизатор производит подключение к источнику данных (например Yammer). Следующим этапом является постраничная обработка данных в соответствии с настройками компании. Тут же будет работать проверка на дубликаты идей. После того, как идея была получена, будет произведена проверка по таким параметрам как:

- тип рабочей сети;
- группа идеи;
- тема идеи;
- дата создания.

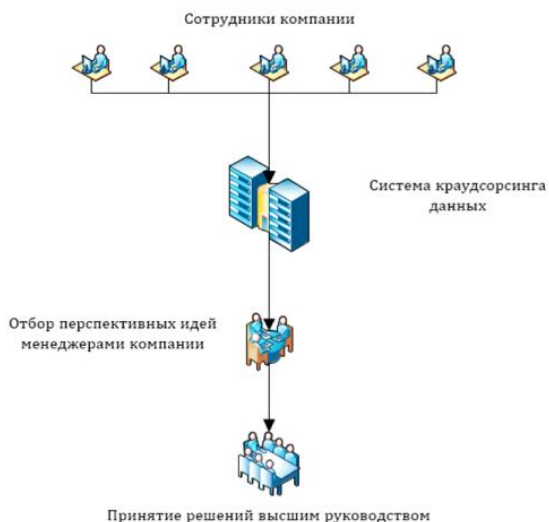


Рисунок 1 – Схема процесса краудсорсинга идей

Если идея будет удовлетворять всем параметрам – идея будет сохранена в базе данных, иначе будет произведен переход к следующей идее в очереди. Когда очередь станет пустой, механизм синхронизации закончит свою работу. Разработанный алгоритм представляет собой агрегацию более низкоуровневых алгоритмов, которые применяются для сбора данных в зависимости от источника. Например, каждая корпоративная социальная сеть или же мессенджер, имеют свой API, в котором модели данных различаются. Сложность задачи состоит в том, чтобы произвести агрегацию данных и представить данные в едином виде.

Передовыми технологиями в данной области являются сверхпроизводительные сервисы Azure [5], служащие для распределения нагрузки в системе. Так, для анализа данных имеет смысл использовать Text Analysis - сервис Azure для качественной оценки данных. Также данный сервис позволяет выделить ключевые слова для каждой выделенной сущности. Полученные данные могут быть использованы при осуществлении аналитики данных.

Кроме того, для области синхронизации с источниками данных актуально применение микро-сервисной архитектуры с элементами событийно-ориентированной модели для максимальной оптимизации приложения. Данная оптимизация касается как скорости работы системы, так и объемов потребляемых ресурсов. К примеру, использование Azure Service Bus позволяет воспользоваться отложенным выполнением задач в системе, а также построить максимально гибкую систему обработки информации. Кроме того, шина сообщений позволяет создать систему с поддержкой распределенной нагрузки, которая будет возникать в случае большого количества данных в источниках.

Применение Azure Service Fabric позволяет добиться высокой производительности системы и гибкого увеличения ресурсов по мере необходимости. Предполагаемая нагрузка системы, как правило, высока ввиду большого количества данных в источнике.

Таким образом, решение проблемы оценки степени инновационности идей можно выполнить с помощью современных технологий. Предлагаемая автоматизированная программная система краудсорсинга идей облегчит поиск перспективных направлений развития компании, аудит рабочих процессов в отделах, а также будет способствовать вовлеченности сотрудников в процесс продвижения компании на рынке конкурентов.

**Список использованных источников:**

1. Crowdsourcing: Why the Power of the Crowd is Driving the Future of Business - Jeff Howe, 2014.
2. Тапскотт Д, Энтони Д. Уильямс. Викиномика. Как массовое сотрудничество изменяет все / Тапскотт Д, Энтони Д. Уильямс. – BestBusinessBooks. – 2009. – 223 с.
3. Crowdsourcing for Dummies – David Alan Grig, 2013.
4. Краудсорсинг: основные площадки [Электронный ресурс]. - <http://www.towave.ru/pub/kraudsorsing-osnovnye-ploshchadki-v-rossii-i-ikh-analogi-na-zapade.html> Дата доступа: 05.04.2017.
5. Azure [Электронный ресурс]. - <https://azure.microsoft.com/en-us/services/> Дата доступа: 05.04.2017.

## МЕТОД ШНОРРА В КРИПТОГРАФИИ

Лапытько А. И., Свиридчик В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Стройникова Е. Д. – ст. преп. кафедры информатики

Данная работа посвящена одному из наиболее эффективных методов аутентификации, а именно методу Шнорра. Для его реализации были разработаны две программы на языках программирования Python (в среде PyCharm) и C++ (в среде Qt). Проведено сравнение метода Шнорра с другими методами аутентификации, изучены сферы их применения.

В наше время, когда цифровые технологии внедряются во все сферы человеческой жизни, людям приходится адаптироваться: переносить личные данные, важную информацию на вычислительные устройства. И вполне естественно скрывать некоторую информацию от отдельных лиц. Для этого в цифровом пространстве используются методы, или схемы, аутентификации. Схема Шнорра была представлена немецким математиком Клаусом Шнорром как видоизменение схем Эль-Гамала и Фиата – Шамира. Основным преимуществом метода Шнорра является меньший размер подписи. После публикации автор метода оформил множество патентов.

Во время разработки биткоина Сатоше Накамото было необходимо выбрать метод реализации цифровой подписи. Первоначально был выбран ECDSA, однако в 2014 г. начал внедряться метод Шнорра.

Большинство алгоритмов криптографии с открытым ключом основывается на таких задачах, как разложение на множители (он же метод факторизации, основанный на сложности разложения на множители большого числа), а также решение дискретного логарифма. Безопасность данного способа шифрования основывается на том, что вычислить  $a^x \pmod p$  легко, например быстрым (бинарным) возведением в степень, а вот вычислить соответствующий логарифм довольно трудно. Наиболее популярными алгоритмами, использующими в своей основе проблему дискретного логарифма в конечном поле, являются:

1. Elgamal (схема Эль-Гамала, 1985 г.) – усовершенствованная версия алгоритма Диффи – Хеллмана, вошедшая в основу большинства последующих схем шифрования.
2. Схема Шнорра (1991 г.) – модификация схем Эль-Гамала и Фиата – Шамира.
3. DSA (Digital Signature Algorithm, 1998 г.), но только для электронной подписи. Так же, как и схема Шнорра, является модификацией схемы Эль-Гамала.
4. ECDSA (Elliptic Curve Digital Signature Algorithm, 1999 г.) – модификация схемы DSA, но основанная на эллиптических кривых. Тем не менее, для криптовалюты схема Шнорра всё ещё является приоритетной.
5. BLS (Боне, Линна, Шахама) – альтернатива подписи Шнорра в криптовалюте.

Для демонстрации схемы Шнорра авторами данной работы были разработаны две программы на языках программирования Python (в среде PyCharm) и C++ (в среде Qt), алгоритм и результат работы которой представлен на рисунке 1.

```
MainWindow
Генерация ключей:
Выбирается простое p= 8191 и простое q= 13 (q|p-1)
Вычисляется g из условия g^q=1(mod p), в данном случае g= 16
Алиса выбирает секретный ключ w= 9 и вычисляет открытый y=g^(-w) (mod p) = 8
Алиса отправляет открытый ключ (p,d,g,y) соответственно равный (8191,13,16,8) , закрытый оставляет у себя - w = 9
Проверка подлинности:
Алиса выбирает случайное число r = 6 и отправляет x=g^r (mod p) = 2048
Боб отправляет Алисе случайное число e = 4
Алиса считает s = (r + w*e) (mod q) = 3 и отправляет s Бобу
Боб вычисляет z=g^s * y^e (mod p) = 2048 и идентифицирует Алису, так как z = x
```

Рисунок 1 – Алгоритм и результат работы программы на языке C++

Разработанные программы доступны по ссылкам:

- 1) <https://github.com/AHTOH2001/MGiA>;
- 2) <https://github.com/sviridchik/shnor/tree/master/sign>.

Алгоритм ECDSA неплох, он выполняет поставленную перед ним задачу, но не более того. К

примеру, нельзя с помощью ECDSA комбинировать ключи и подписи и нельзя одновременно подтвердить каждую подпись, что особенно полезно в транзакциях с мультиподписью.

Алгоритм Шнорра в этом плане более эффективен: он способен скомбинировать набор подписей и ключей в один блок и даёт возможность их проверить гораздо быстрее, чем каждый ключ и подпись в отдельности. Но и он имеет некоторые недостатки, такие как:

- создание нескольких подписей для каждого блока;
- неудобство продолжительного хранения блока, т. к. используются несколько коммуникационных каналов для подписи;
- многократное генерирование случайного числа при комбинировании вместо того, чтобы выбрать одну случайную точку, как в ECDSA;
- использование пары простых чисел.

Алгоритм BLS способен решить все вышеперечисленные проблемы:

- используется одно простое число для подписи;
- генератор случайных чисел вообще не используется;
- блок можно представить одной подписью;
- нет необходимости в нескольких каналах для подписи.

Это результат того, что в алгоритмах ECDSA и Шнорра хешируется сообщение и используется этот же хеш, представляющий собой число, в подписи. А в алгоритме BLS сразу получается хеш, представляющий собой эллиптическую кривую, такую же, как и те, которые используются в криптовалюте. Таким образом, несмотря на все преимущества подписи BLS он может использоваться только в криптовалюте, т. е. он не так гибок, как алгоритм подписи Шнорра.

Схема Шнорра лежит в основе стандарта Республики Беларусь СТБ 1176.2-99 и южнокорейских стандартов KCDSA и EC-KCDSA.

**Список использованных источников:**

1. Стройникова, Е. Д. Основы прикладной алгебры : учеб.-метод. пособие / Е. Д. Стройникова. – Минск : БГУИР, 2010. – 120 с.
2. Википедия, схема Шнорра [Электронный ресурс]. – Режим доступа : [https://ru.m.wikipedia.org/wiki/Схема\\_Шнорра](https://ru.m.wikipedia.org/wiki/Схема_Шнорра).
3. Лекция 3: Цифровая подпись [Электронный ресурс]. – Режим доступа : <https://www.intuit.ru/studies/courses/553/409/lecture/9379?page=5>.
4. Лекция 3: Цифровая подпись, страница 5 [Электронный ресурс]. – Режим доступа : <https://www.intuit.ru/studies/curriculums/15720/courses/409/lecture/9379?page=5>.
5. Глава 21 Схемы идентификации [Электронный ресурс]. – Режим доступа : <https://studfile.net/preview/1826159/page/38/>.
6. Схемы идентификации 21.1 FEIGE-FIAT-SHAMIR [Электронный ресурс]. – Режим доступа : <http://wclipart.narod.ru/doc-1/kind-crip23-1-pag.html>.

## ОТСЛЕЖИВАНИЕ ЛИЦ НА ВИДЕО С ИСПОЛЬЗОВАНИЕМ КЛАССИЧЕСКИХ И НЕЙРОСЕТЕВЫХ ПОДХОДОВ В КОМПЬЮТЕРНОМ ЗРЕНИИ

*Латушкин Н.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Сиротко С.И. – к.т.н., доцент*

В докладе приводится описание инструмента позволяющего вести работу с лицами на видео. Также описываются принятые решения и модули необходимые для построения системы.

В современном мире одним из наиболее популярных типов данных являются видеоданные. Поэтому обработка видео является частой и необходимой задачей. Однако обработка такого типа данных требует большого количества времени и ресурсов. Например, если мы хотим обработать 1 минуту видео, в случае отсутствия специальных программных средств для этих целей, мы будем вынуждены обработать целых 1800 кадров (как правило одна секунда видео содержит 30 кадров). Одной из наиболее часто решаемых задач в контексте обработки видео является задача обнаружения и распознавания людей.

Таким образом система для автоматического отслеживания лиц на видео должна содержать в себе следующие логические компоненты: детекция лиц, распознавание лиц, трекер лиц, модуль анализа и объединения информации, а также удобный графический интерфейс для работы с видео в случае, когда нам понадобится исправлять ошибки автоматического аннотирования видео.

Детектирование лиц необходимо для нахождения всех лиц на кадрах, а также их последующего сопоставления с людьми. В качестве решения для этой задачи будем использовать нейросетевой детектор на основе нейронной сети Retina с использованием в качестве backbone архитектуры

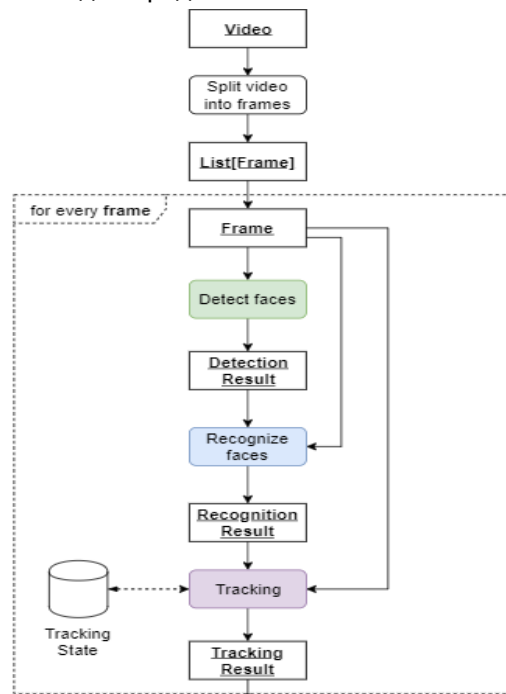


Resnet50. В качестве входа она будет получать, по очереди, каждый кадр из видео. Таким образом, после работы этого модуля мы получим все координаты рамок обрамляющих лица.

Распознавание лиц используется для агрегации рамок, полученных в результате работы модуля детектирования, в коллекции, внутри которых содержатся рамки одного и того же человека. Для этих целей мы будем использовать нейронную сеть по типу архитектуры ArcFace с backbone Resnet50. В качестве входных данных будем использовать вырезанные по координатам рамок детекции изображения. Их также необходимо выровнять и привести к одному размеру. На выходе получим набор дескрипторов, соответствующих каждой из поданных на вход рамок. Данные дескриптора будут обладать следующим свойством: если посчитать метрику косинусной схожести для двух дескрипторов, то у одинаковых лиц она будет высокой, а у разных низкой.

Трекинг лиц используется для устранения минусов работы модулей детекции и распознавания лиц. Иногда модуль детекции лиц может ошибаться и не детектировать все лица на кадре, тогда в случае наличия этого человека на других кадрах видео, с помощью трекинга мы имеем шанс найти этого человека на всех кадрах, на которых он присутствует. В качестве трекера будет использоваться SiamRpn сети, принимающие на вход все ту же рамку детекции.

Общая схема обработки видео представлена ниже:



Общая схема обработки видео

**Список использованных источников:**

1. Examples of visual object tracking Электронный ресурс / Режим доступа: <https://pt.coursera.org/lecture/deep-learning-in-computer-vision/examples-of-visual-object-tracking-methods-NmYR4>
2. Face detection Электронный ресурс / Режим доступа: <https://arxiv.org/abs/1905.01585>
3. Face recognition Электронный ресурс / Режим доступа: <https://arxiv.org/abs/1801.07698>

## РАСПОЗНАВАНИЕ ЛИЦ НА ИЗОБРАЖЕНИЯХ ПЛОХОГО КАЧЕСТВА С ИСПОЛЬЗОВАНИЕМ НЕЙРОСЕТЕВЫХ АЛГОРИТМОВ

*Латушкин Н.С., Сухов Н.Ю., Козак А.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Сиротко С.И. – к.т.н., доцент*

Наша работа представляет собой исследование различных подходов к решению проблемы распознавания лиц по низкокачественным изображениям. Также мы провели анализ потенциальных проблем и их решений, которые могут возникнуть в ходе решения поставленной задачи.

Распознавание лиц на протяжении десятилетий было очень активной областью

компьютерного зрения. Существует много различных баз данных с изображениями лиц, связанных с ними конкурсов и программ оценки, которые поощряют инновации: создание более мощной технологии распознавания лиц с многообещающими (потенциально лучшими) результатами. В последние годы наблюдается огромное улучшение в качестве распознавания лиц с использованием сложных глубоких архитектур нейронных сетей, тренированных на миллионах изображений лиц. Не смотря на то, что существуют алгоритмы распознавания лиц, которые могут эффективно работать с кооперативными субъектами в контролируемых средах и с некоторыми неограниченными состояниями, распознавание лиц на низкокачественных изображениях далеко от идеала.

Низкокачественные изображения лиц в основном получаются с помощью четырёх процессов деградаций, применяемых к изображениям высокого качества:

- Размытие, связанное с нефокусировкой объектов, наложением, относительным движением объекта-камеры, атмосферной турбулентностью и т. д.
- Низкое разрешение, вызванное большим расстоянием до камеры и (или) датчиком камеры с низким пространственным разрешением.
- Артефакты вызванные низкоскоростным сжатием или другими причинами.
- Условиями приема сигнала, которые добавляют в изображение шум (например, низкий уровень освещённости).

В действительности, низкокачественные изображения могут содержать все четыре процесса деградации, тогда как обычно в синтетических низкокачественных изображениях моделируется только один процесс, например, изображения с низким разрешением формируются путем нисходящей выборки и добавлением искусственно размытости, созданной низкочастотной фильтрацией высококачественных изображений лиц. В нашей работе были рассмотрены методы, которые позволяют распознавать лица с учетом этих процессов деградации. Таким образом, термин низкокачественное изображение может включать в себя любой из упомянутых процессов деградации, а под распознаванием далее будет пониматься распознавание низкокачественных изображений лиц в целом.

Недавние исследования позволяют решать существенные проблемы, связанные с эффективностью распознавания лиц, возникающие из-за низкого качества данных, полученных от зашумленных наблюдений и аналогичных деградационных камерных настроек. Полезность автоматического распознавания лиц в данных камер видеонаблюдений часто мотивируется соображениями общественной безопасностью как в государственном, так и в частном секторах. Задачи распознавания в данном случае могут включать в себя:

- Идентификация по списку наблюдений — это задача определения того, соответствует ли обнаруженное лицо человека лицу из списка интересующих нас людей.
- Повторная идентификация (реидентификация) — определение, является ли человек, чьё лицо было захвачено в одно время одной камерой, человеком, чьё лицо было снято в другое время и (или) другой камерой.

В нашей работе мы исследовали и рассмотрели следующие технологии решения этих задач:

- Улучшение качества изображения с помощью нейронной сети (Super-resolution).
- Использование робастных признаков для низкокачественных изображений.
- Подавление размытия.
- Использование единого пространства для робастных признаков к высокому и низкому разрешению.

В нашей работе мы всесторонне обобщили и рассмотрели различные решения проблемы распознавания лиц по низкокачественным изображениям за последние десятилетия. Также мы раскрыли будущие проблемы, которые могут возникнуть в ходе решения поставленной задачи. Мы пришли к выводу, что выбор методов улучшения распознавания лиц на низкокачественных изображениях является важной темой современных исследований. Среди нами изученных методов, глубокое обучение позволяет очень эффективно бороться с некоторыми проблемами (например, низкая освещённость, расплывчатость, нестандартизированные позы и т. д.) и всё же плохо в других случаях (например, низкое разрешение, сжатие и т. д.). Также мы выяснили, что высокий уровень интереса в научном мире к распознаванию лиц на низкокачественных изображениях обусловлен такими перспективными областями применения, как криминалистика, видеонаблюдение, общественная безопасность и другие.

**Список использованных источников:**

1. Face Recognition [Электронный ресурс] – Режим доступа: <https://ru.coursera.org/lecture/convolutional-neural-networks/what-is-face-recognition-IUBYU>.
2. ArcFace [Электронный ресурс] – Режим доступа: <https://arxiv.org/abs/1801.07698>.
3. Super Resolution [Электронный ресурс] – Режим доступа: <https://arxiv.org/abs/1912.01839>.

# ВИЗУАЛИЗАТОР ПРОСТРАНСТВЕННО-ВРЕМЕННЫХ ФУНКЦИЙ

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Резунов М. В.

Теслюк В. Н. – кандидат физико-математических наук, доцент

В докладе рассматриваются задачи построения статических и динамических графиков функций; нахождения геометрических тел в множестве многоугольников, задачи вхождения точки в геометрическую фигуру на плоскости и в геометрическое тело в пространстве; создания интерактивного оконного графического приложения. В докладе формулируется суть задач, рассматривается алгоритм решения, анализируются и сравниваются результаты выполнения с аналогами.

Научные исследования утверждают, что 90% воспринимаемой информации — зрительная. Визуализация — это представление различной информации в удобном для зрительного наблюдения и анализа. Позволяет быстрее и эффективнее донести идеи и мысли. [1]

Задачи доклада:

- 1) Построение и отображение статических и динамических графиков функций.  
Для расчета значений функций используются методы численного анализа и преобразование функций из инфиксной нотации в обратную польскую. [2,3]
- 2) Нахождение геометрических тел в заданном множестве многоугольников в пространстве.
- 3) Определение положения точки относительно геометрической фигуры на плоскости.
- 4) Определение положения точки относительно геометрического тела в пространстве.
- 5) Создать приложения, выполняющие поставленные задачи.

Приложение отвечает поставленным задачам и также позволяет:

- сохранять и загружать построенные графики, использовать заранее построенные графики;
- отображать динамические функции в любой момент времени из заданного диапазона времени;
- загружать 3D объекты из файла и отображать их;
- визуализировать результат вхождения точки в геометрическое тело полностью и пошагово.

На практике приложение используется в различных областях науки и техники:

- естествознание: моделирование различных процессов от распределения температуры до образования звезды, география, экология;
- математика: построение графиков, мат. моделирование, мат. анализ, аналитическая геометрия;
- медицина: рентгенография, МРТ, ультразвук, компьютерная томография, оксиметрия и др.

Результатом является график функции, построенный из полигонов на точках матрицы по заданной цветовой схеме (рисунки 1, 2), а также минимальные и максимальные значения функций.

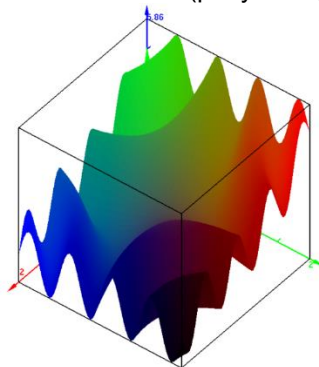


Рисунок 1.  $z = \sin(3.14*x*y)/0.5 + x^2 - 2.71*x$

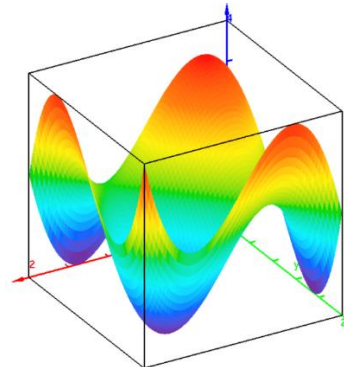


Рисунок 2.  $z = x^3 - 3*x + y^3 - 3*y$

#### Список использованных источников:

1. Материал сайта [Электронный ресурс]: <https://habr.com/ru/company/devexpress/blog/240325/>
2. Материал сайта [Электронный ресурс]: [https://ru.wikipedia.org/wiki/Обратная\\_польская\\_запись](https://ru.wikipedia.org/wiki/Обратная_польская_запись)
3. Методы численного анализа математических моделей. / М. Галанин, Е. Савенков. 2015. — 593 с.

## ОЦЕНКА КАЧЕСТВА УЧЕБНОГО МАТЕРИАЛА ПОСРЕДСТВОМ МЕТОДИК АНАЛИЗА ДАННЫХ

Саскевич А.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Стержанов М.В. – канд. тех. наук

В данной работе приведены результаты исследований по работе над оценкой качества учебного материала на примере обработки данных для интерактивного курса по изучению иностранного языка. В качестве основных инструментов применены скрапинг-системы и модели обработки натурального языка на языке Python.

При создании учебного материала применяется как ручной способ сбора и обработки материала, так и активно развивающиеся методики генерации учебного материала посредством инструментов машинного обучения и обработки больших объемов информации. В обоих случаях оценка качества может производиться как вручную, так и автоматически. Например, можно оценить сложность примера для изучающего курс высшей математики как количество математических операций, которые необходимо выполнить для решения задачи. С другой стороны, посредством разнообразных механизмов возможно в автоматическом режиме, с учетом каких-то скрытых и недоступных наблюдателю факторов оценить материал в достаточно крупном объеме. Подобные подходы уже применяются, например, в системах, позволяющих адаптивно формировать учебный материал [1].

В качестве данной работы проводилась разработка метрик, посредством которых возможна оценка учебного материала, сгенерированного автоматизированной системой для интерактивного курса иностранного языка. Условно процесс можно разделить на несколько этапов:

1. Скрапинг данных посредством любой доступной скрапинг-утилиты, например, библиотека Scrapy;
2. Извлечение текстовых и/или графических данных с целью формирования базы данных учебного материала;
3. Формирование атомарных учебных заданий (например, перевод фразы или перевод слова), которое может быть выполнено посредством любого модуля обработки естественного языка;
4. Расчет метрик для атомарных учебных заданий по некоторой методике [2].

Для расчета метрик для учебного языкового курса произведена следующая процедура:

1. Разделение текста на токены – отдельные слова и знаки препинания;
2. Подсчет частоты вхождения подобных токенов и их тип (существительные, глаголы и т.д.);
3. Соотнесение частоты и типа с некоторым взвешенным рейтингом, например, по формуле взвешенного среднего арифметического:

$$\frac{\sum x_i w_i}{\sum w_i} \quad (1),$$

где  $x_i$  – категоризированное значение признака,  $w_i$  – частота вхождения данного признака в выборку.

Затем задания упорядочены в порядке увеличения частоты, вычисленной на основе слов и фраз, входящих в их состав. Следующим этапом является формирование графа зависимостей, который призван показать, какие слова и фразы зависят от предварительного изучения других слов и фраз. Пример такого графа представлен на рисунке 1.

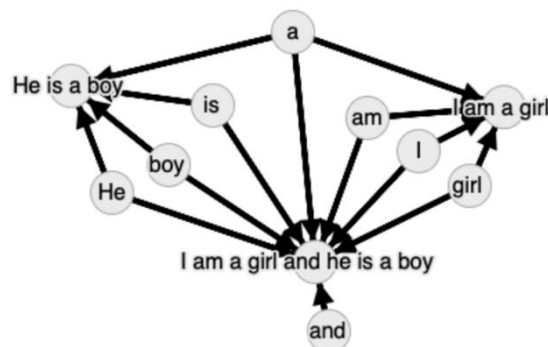


Рисунок 1 – Граф зависимостей слов при изучении

На такой граф могут быть перенесены различные варианты алгоритмов на графах, которые

могут помочь либо улучшить и дополнить метрики качества, описанные на собранных данных, либо заменить их альтернативным вариантом. В качестве применения подойдет множество различных классических алгоритмов на графах [3].

Топологическая сортировка позволяет выявить последовательность материала для изучения. Адаптация задачи под транспортную задачу с максимальным потоком позволяет определить точки входа для учащегося. Например, вершины, в которых отсутствуют входящие ребра, являются начальной базой для изучения.

На следующем этапе возможно соотнести полученные вершины графа с рассчитанными ранее значениями среднего взвешенного арифметического, позволяя упорядочить и выбрать в качестве стартовых элементов не просто слова, которые содержат в построенном графе исходящие ребра, но и которые являются, например, более популярными в своей категории – например, существительные и местоимения.

Для создания определенной сложности учебный материал можно наполнять словами, которые не были изучены ранее и отдельно, повышая рейтинг таких заданий. В качестве примера был собран набор слов и фраз для французского языка. На рисунке 2 представлены графики, на которых представлено, что формирование учебного материала в огромной степени зависит от включения материала, который будет предложен к изучению в контексте – например, внедрение неизвестных одного-двух слов в предложение, в котором учащийся уже знает остальные слова. Данный подход позволяет охватить до 80% доступного учебного материала, опираясь только на 20% изучаемого материала:

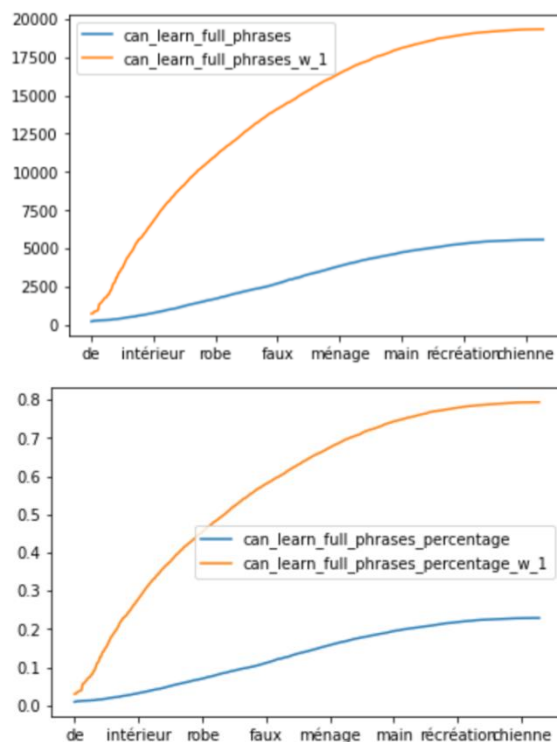


Рисунок 2 – Зависимость охвата учебного материала от количества изучаемого материала

Таким образом, применяя и комбинируя разнообразные статистические метрики, можно выполнить предварительную оценку учебного материала, в частности, сгенерированного посредством автоматизированных информационных систем. Предварительная оценка позволяет определить материал достаточного уровня, который в дальнейшем может быть протестирован на реальных учащихся.

**Список использованных источников:**

1. Тархов С. В. Адаптивное электронное обучение и оценка его эффективности // Открытое образование. 2005. №5. – С. 37-47.
2. Кузнецова Ирина Александровна Оценка качества систем дистанционного обучения // Вестник евразийской науки. 2011. №2 (7).
3. Неверов П. А. Использование графов для структурирования и дальнейшего совершенствования учебного материала // Известия РГПУ им. А.И. Герцена. 2008. №63-2.

## НЕЙРОСЕТЕВАЯ СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЯ О ВЫДАЧЕ КРЕДИТА

Сенькович Д.С.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Жвакина А.В. – канд. техн. наук, доцент

Обсуждаются различные подходы к анализу информации о клиентах банка с целью принятия решения о выдаче кредита. Исследованы различные способы моделирования данной задачи, оценена их точность и время получения результатов.

При выдаче кредите банком требуется некоторым образом оценить платежеспособность клиента. Один из способов - кредитный скоринг [1].

Кредитный скоринг является трудоемким и в некотором смысле недостаточно точным процессом, так как требует формирования правил подсчета скоринга человеком. Такие правила включают в себя не только такие очевидные как уровень дохода, но и более сложные демографические и социальные аспекты жизни клиента. Но, ввиду наличия человеческого фактора, многие показательные мелочи могут быть не замечены и не учтены. Например, порой банку выгодно выдать кредит зная, что он будет просрочен, в случае чего банк сможет получить прибыль с штрафов просрочки [2, 3].

Автоматическая система принятия решения о выдаче кредита могла бы анализировать гораздо большие объемы информации, вместе с тем находя более сложные зависимости и выделяя наиболее важные виды данных о клиенте, характеризующие его как потенциально успешного заемщика.

В работе рассматривались различные модели машинного обучения для построения такого решения на основе данных 1000 клиентов и решения банка: выдавать кредит или нет.

Исследовалась нейронная сеть patternnet, которая при проверке на тестовых данных показала точность 96,7%, общая точность – 92,4.

При экспериментах с сетью lvqnet (learning vector quantization neural network), получено более длительное время обучения (8 сек.) и значения точности хуже (91%), что демонстрирует нецелесообразность использования данной сети для решения поставленной задачи.

Исследовались и классические модели машинного обучения с различными архитектурами многослойных перцептронов. В таблице 1 ниже представлены исследованные модели с наилучшими параметрами и наилучшие результаты, соответствующие данным моделям:

Таблица 1 – Различные модели машинного обучения с наилучшими параметрами

Модель	Параметры	Время обучения всех исследованных моделей	Время обучения лучшей модели	F1 оценка	Точность
Логистическая регрессия	Алгоритм оптимизации lbfgs, L2 регуляризация, C 0.23	40.94 секунд	0.0149 секунды	0.9595015576	90.5%
SVM	Ядро sigmoid, C 0.17	3.02 секунды	0.005 секунды	0.9565217391	91.1%
Случайный лес	265 базовых алгоритмов, построение деревьев до конца	32.95 секунды	0.34 секунды	0.9620253165	91.5%
Нейронная сеть	Оптимизатор adam, количество слоев 1, количество нейронов 61, dropout rate 0.4625, коэффициент обучения 0.0741, обучение в течение 50 эпох	23611.46 секунд ≈ 6.5 часов	4.925 секунды	0.9565217391	95.5%

В работе рассмотрены следующие модели машинного обучения: логистическая регрессия, метод опорных векторов (SVM), алгоритм случайного леса (random forest), нейронные сети. Наилучшие результаты удалось получить с использованием нейронных сетей.

**Список использованных источников:**

1. Что такое кредитный скоринг? [Электронный ресурс]. – Режим доступа: <https://creditregister.by/Help/WhatsCreditScoring>. - Дата доступа: 16.02.2020.
2. Как банки принимают решение о выдаче кредита [Электронный ресурс]. – Режим доступа: [https://mycreditinfo.ru/kak\\_banki\\_prinimaut\\_reshenie\\_o\\_vydache\\_kredita](https://mycreditinfo.ru/kak_banki_prinimaut_reshenie_o_vydache_kredita). - Дата доступа: 16.02.2020
3. Как банки проверяют заемщиков [Электронный ресурс]. – Режим доступа: <https://www.sravni.ru/enciklopediya/info/kak-banki-proverjajut-zajomshhikov>. - Дата доступа: 16.02.2020

## АНАЛИЗ ЭМОЦИЙ ЧЕЛОВЕКА ПО ПОРТРЕТНОМУ СНИМКУ

*Совпель Д. С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Калугина М. А. – канд. физ.-мат. наук, доцент*

Рассмотрены алгоритмы построения моделей, основанных на применении сверточных нейронных систем, для классификации эмоций портретного снимка человека. Описана работа моделей, условие их применения в зависимости от выбранного подхода к построению алгоритма, представлены результаты и возможные перспективы улучшения модели.

Многие задачи искусственного интеллекта и машинного обучения сводятся к автоматизации обработки большого количества данных. Одной из таких проблем является определение эмоций на основе его снимка или видеофрагмента. Такая система имеет множество сфер применения: система наблюдения за озлобленными гражданами позволит предотвратить возможное преступление, HR в компаниях смогут определить незаинтересованных кандидатов на работу. Не всегда возможно выполнять такие задачи “вручную”, так как объем данных для обработки увеличивается, и скорость классификации эмоций человеком несоизмерима с количеством данных.

В основе всех алгоритмов была выбрана классическая структура построения моделей обработки изображений – использование сверточных нейронных сетей. Для таких задач возможен выбор нескольких подходов: использование уже тренированной архитектуры и разработка и обучение собственной архитектуры. Поэтому было выбрано два основных алгоритма: модель на базе архитектуры ResNet и собственная модель. Для построения модели был использован датасет FER 2013 [1]. Он состоит из данных, разбитых на множества для тренировки и валидации (теста) модели. Каждая картинка представляет собой черно-белое изображение 48 на 48 пикселей. Все изображения разделены на 7 классов: злорада, отвращение, испуг, счастье, грусть, удивление и спокойствие. Для того чтобы использовать такие изображения для обучения, было увеличено количество каналов (так как сверточные сети используют цветные изображения с 3 цветовыми каналами) путем копирования изображения 3 раза.

Модель на базе ResNet [2] включает в себя дообученные слои модели ResNet и слоев для дообучения сети до конкретной задачи классификации. Такой подход позволяет, во-первых, использовать уже предтренированные слои для выделения общих компонент изображения, так как первые слои данной модели выполняют именно эту задачу, во-вторых, уменьшить вычислительную сложность для тренировки модели под конкретную задачу при помощи добавленных слоев. Поэтому к архитектуре ResNet был добавлен один слой субдискретизации и 3 полносвязных слоя с регуляризацией дроп-аутом. Также часть последних слоев исходной архитектуры была также использована для дообучения под классификацию эмоций.

Вторая модель построена следующим образом: три слоя, состоящие из сверточного слоя нормализации по батчу, функции активации и слоя максимальной субдискретизации, а затем 3 полносвязных слоя с дроп-аутом для регуляризации. Между слоями с параметрами для обучения расположены слои для регуляризации. Это позволяет предотвратить переобучение модели на относительно маленьком датасете.

Для реализации моделей использовался язык Python 3.8, библиотека Pandas для обработки датасета FER 2013 и библиотека для построения глубоких сетей Keras. Эти инструменты позволяют наиболее быстро реализовывать, тренировать и оценивать модели нейронных сетей. Модель с данной архитектурой показала точность в 57% на валидационном множестве (рис.1).

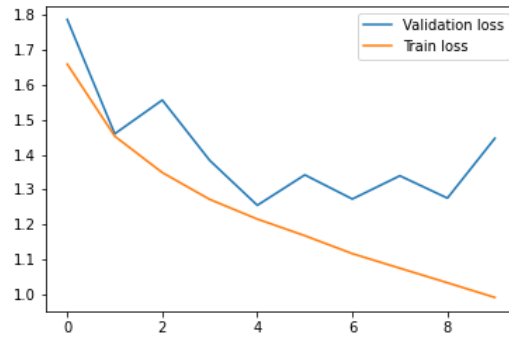


Рисунок 1 - график скорости обучения модели с разработанной архитектурой

Модель с архитектурой на базе ResNet показала точность в 68% на валидационном множестве (рис.2).

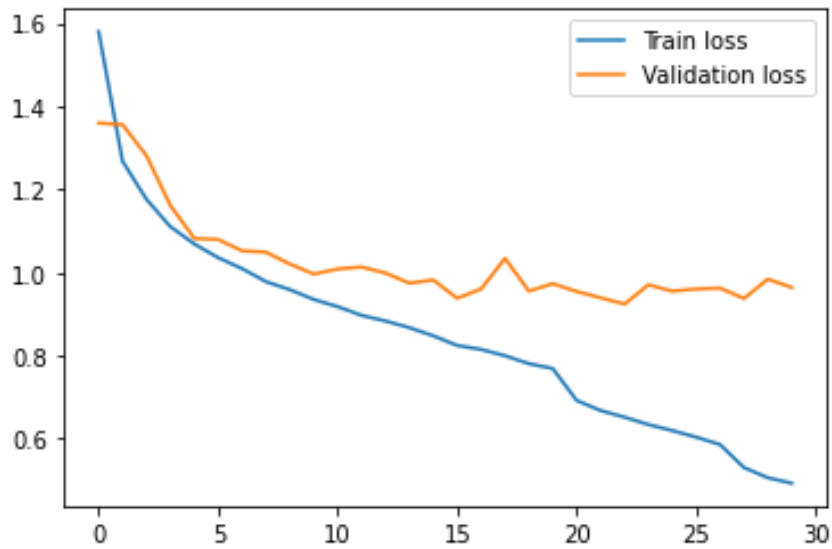


Рисунок 2 - график скорости обучения модели с архитектурой на основе ResNet



```
(dl_py38) $ python predict.py -i angry_face.jpg -m resnet18
angry: 99.36%
disgust: 0.00%
fear: 0.26%
happy: 0.12%
sad: 0.26%
surprise: 0.00%
neutral: 0.00%
```

Рисунок 3 - пример работы алгоритма на основе ResNet



Таким образом, построенные модели могут использоваться для классификации снимков человека. Данные модели можно использовать в связке с алгоритмами обнаружения лица, которые позволяют выделить лицо человека, для систем непрерывной видеосъемки, усредняя множество кадров и получая одно изображение для классификации (рис.3). В результате можно построить систему непрерывного наблюдения за эмоциями человека.

**Список использованных источников:**

1. Dataset fer2013 // [Электронный ресурс] <https://www.kaggle.com/ahmedmoorsy/facial-expression>
2. Deep Residual Learning for Image Recognition / K. He [et al.] // arXiv:1512.03385v1 [cs.CV], 2015.

## **ОБНАРУЖЕНИЕ РАКА ЛЁГКОГО ПО СНИМКАМ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ**

*Сухов Н.Ю.*

*Белорусский государственный университет информатики и радиоэлектроники*

*г. Минск, Республика Беларусь*

*Калугина М.А. – к.ф.-м.н., доцент*

В докладе рассматривается построение автоматизированной системы определения рака легкого человека, а также описывается инструмент, показывающий злокачественные образования на снимке.

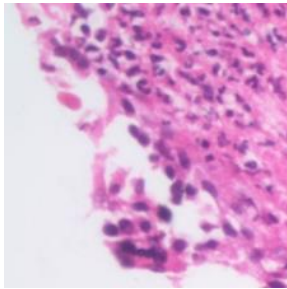
Рак легких является основной причиной смерти от рака в Соединенных Штатах и во всем мире. Прогрессирование заболевания и ответ на лечение при раке легких широко варьируется среди пациентов. Следовательно, точный диагноз имеет решающее значение при выборе и планировании лечения для каждого пациента с раком легких. Микроскопическое исследование тканей легкого остается важным этапом диагностики рака. Это требует, чтобы патолог распознал тонкие гистопатологические паттерны на очень сложных изображениях тканей. Этот процесс занимает много времени и является субъективным. Также, чтобы поставить верный диагноз, требуются высококвалифицированные специалисты. Поэтому построение автоматизированной системы является крайне важной задачей. На сегодняшний день с применением алгоритмов машинного и глубинного обучения такие задачи решаются весьма успешно.

В имеющемся датасете имеется 15000 гистологических снимков лёгкого. 5000 из них представляют из себя доброкачественную ткань, а 10000 имеют злокачественные образования. Целью является построить бинарный классификатор, где класс 0 свидетельствует о здоровой ткани, а класс 1 - о раке. Классификатор был построен на основе сверточной нейронной сети, так как данный тип нейронных сетей за последних несколько лет себя очень хорошо зарекомендовал в задачах компьютерного зрения. Работа сверточной нейронной сети обычно интерпретируется как переход от конкретных особенностей изображения к более абстрактным деталям, и далее к еще более абстрактным деталям вплоть до выделения понятий высокого уровня. При этом сеть самонастраивается и вырабатывает сама необходимую иерархию абстрактных признаков (последовательности карт признаков), фильтруя маловажные детали и выделяя существенное. В данной работе использовалась собственно написанная нейронная сеть, которая представляет из себя чередующиеся сверточные слои с функцией активации RELU и слои пулинга (для уменьшения пространственной размерности).

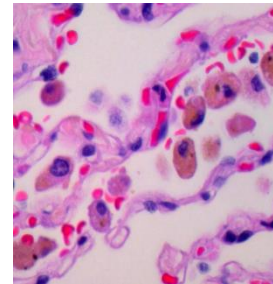
Сеть, описанная выше, показала очень хорошие результаты, однако приблизиться к результату врачей не удалось. Далее я стал использовать предобученную нейронную сеть. Обычно используют сети предобученные на датасете ImageNet, которая содержит более 15 миллионов изображений. Однако, в рамках данной задачи, использование этого датасета только ухудшает предыдущий алгоритм, так как он состоит из изображений кошек, собак, домов и т. д., а в нашем наборе очень специфические данные. Среди долгих поисков данных выбор пал на гистологические снимки толстой кишки. В этих данных есть общее с нашими данными для лёгких: много похожих клеток и структур.

Ещё в помощь для построения хорошей классификации пришёл алгоритм кластеризации KMeans (исследовались и другие алгоритмы кластеризации, например, DBSCAN, однако, особого прироста в качестве они не давали, а обучались очень долго, что также повлияло на время выдачи предсказаний всей системы на тестовых данных). Оказалось, что “прогнав” через кластеризацию снимки лёгких, у нас неплохо отделились кластера с различными клетками и межклеточное пространство. Обучаясь на кластеризованном датасете, снизилось немного переобучение, так как снимки могут приходиться в немного различной цветовой гамме. Финальное решение было выбрано

такое: предобученная сверточная нейронная сеть на снимках толстой кишки и дообучение ее на кластеризованных изображениях лёгких, “разморозив” три последних сверточных слоя. Также была использована функция потерь, которая сильнее штрафует за ложно-отрицательные результаты, так как поставить диагноз рака, когда его на самом деле нет, не так опасно, если сказать, что его нет, когда он есть. Это решение достигло качества примерно 97%, когда качество врачей составляет примерно 95 процентов. Таким образом, удалось построить систему, определяющую рак лёгких не хуже, чем врач-специалист.



Снимок толстой кишки



Снимок легкого

Также, чтобы врачу-специалисту можно было понять, почему алгоритм принял то или иное решение, была сделана подсветка областей, смотря на которые, классификатор принял решение, что изображение содержит злокачественное образование. Это было сделано с помощью сегментационной сети Deeolabv3+, которая на данный момент является лучшей сетью для сегментации изображений и в данной задаче также показала очень хорошие результаты.

**Список использованных источников:**

1. Artificial Intelligence in Lung Cancer Pathology Image Analysis [Электронный ресурс]. – Режим доступа: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6895901/>.
2. Классификация рака лёгких [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/445054/>.

## ИСКУССТВЕННЫЙ НЕЙРОН

*Сычёв А.Ю., Стаселько И.Д., Аниховский М.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Стержанов М.В. – к.ф.-м.н., доцент*

К нейронным сетям с обратными связями относят рекуррентные нейронные сети. Рекуррентные нейронные сети показывают хорошие результаты во многих задачах связанных с классификацией последовательностей, например, распознавание речи и подавление шума, более того они имеют одни из лучших результатов на данный момент в некоторых задачах связанных с обработкой текста на естественном языке, например, в задаче классификации текста.

Математическая модель нейрона была разработана в 1950-х годах ученым Франком Розенблаттом, который был вдохновлен работой Питтса и Маккалока. Разработанная модель была названа перцептроном[1].

Искусственный нейрон имеет несколько входов и один выход. Для вычисления выхода каждый вход нейрона умножается на вес, аналогом которого является сила синапса. Далее для вычисления уровня активации полученные значения суммируются.

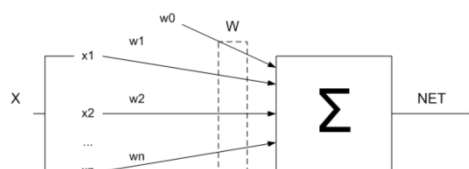


Рис. 1 — Модель искусственного нейрона

На рисунке 1 показана модель искусственного нейрона. Входные значения обозначены  $x_1 \dots x_n$ . Это аналог тех сигналов, что приходят на вход биологических нейронов.  $W$  — веса (сила синаптической связи)[2]. Блок, который суммирует взвешенные входы является аналогом тела биологического нейрона. Математически это можно выразить следующей формулой:

$$NET = \sum_{i=1}^n w_i x_i + w_0$$

Далее выходной сигнал NET пропускается через функцию активации. Модель нейрона с функцией активации показана на рисунке 2.

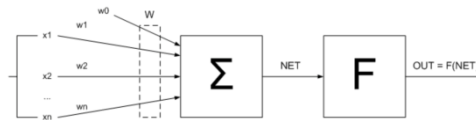


Рис. 2 — Модель искусственного нейрона с функцией активации

**Список использованных источников:**

1. Elman J.L. Finding structure in time // Cognitive science, 1990. Vol. 14, no. 2. P. 179 – 211.
2. Lai S., Xu L., Liu K., Zhao J. Recurrent Convolutional Neural Networks for Text Classification. // AAAI, 2015. P. 2267 – 2273.

## РЕКУРРЕНТНЫЕ НЕЙРОННЫЕ СЕТИ

Сычѳв А.Ю., Стаселько И.Д., Аниховский М.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Стержанов М.В. – к.ф.-м.н., доцент

К нейронным сетям с обратными связями относят рекуррентные нейронные сети. Рекуррентные нейронные сети показывают хорошие результаты во многих задачах связанных с классификацией последовательностей, например, распознавание речи и подавление шума, более того сети имеют одни из лучших результатов на данный момент в некоторых задачах связанных с обработкой текста на естественном языке, например, в задаче классификации текста. Каждая цепочка вычисляет ошибку как суммарную девиацию по выходным сигналам сети. Если имеется набор образцов, ошибка вычисляется с учётом ошибок каждого отдельного образца.

Рекуррентные нейронные сети — подкласс нейронных сетей с обратными связями, которые используют предыдущие состояния сети для вычисления текущего. Сеть строится из узлов, каждый из которых соединѳн со всеми другими узлами. У каждого нейрона порог активации меняется со временем и является вещественным числом. Каждое соединение имеет переменный вещественный вес[1]. Узлы разделяются на входные, выходные и скрытые. Простейший пример такой сети проиллюстрирован на рисунке 1. Такая нейронная сеть имеет обратные связи. Данное отличительное свойство сетей позволяет провести аналогию с биологическим мозгом, так как он тоже способен сохранять свои состояния и реагировать на события, опираясь на предыдущий опыт.

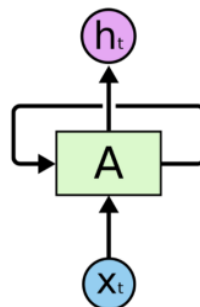


Рис. 1 - Архитектура рекуррентной нейронной сети с одним входным нейроном, одним скрытым нейроном и одним выходным нейроном

Основной класс задач, решаемый такими сетями - это обработка корпусов текстов, т.к. такие сети специально сделаны для того, чтобы обрабатывать последовательности данных неизвестной длины [2]. Каждая такая сеть имеет в себе особую структуру, похожую на звенья или модули, идущие один за другим. И каждый такой модуль представляет собой слой, с какой-либо выходной функцией активации. Кроме этого, такая сеть может выполнять задачу предсказания слова в предложении, так как способна понимать контекст предложения. Рекуррентная нейронная может обрабатывать не только тексты, но и временные ряды, так как может обрабатывать их в двух направлениях. Способ обучения такой сети очень схож с обучением обычной нейронной сети. Разница лишь в том, что рекуррентная сеть требует использования метода обратного распространения ошибки во времени вместо привычного обратного распространения ошибки. Как следствие, на рекуррентные сети можно применять те же самые методы для решения проблемы переобучения, как и с обычными сетями.

**Список использованных источников:**

1. Elman J.L. Finding structure in time // Cognitive science, 1990. Vol. 14, no. 2. P. 179 – 211.
2. Lai S., Xu L., Liu K., Zhao J. Recurrent Convolutional Neural Networks for Text Classification. // AAAI, 2015. P. 2267 – 2273.

## **ОЦЕНКА ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ МОДЕЛЕЙ КЛАССИЧЕСКОГО МАШИННОГО ОБУЧЕНИЯ ДЛЯ АНАЛИЗА ТОНАЛЬНОСТИ РЕЧЕВЫХ СИГНАЛОВ ЧЕЛОВЕКА**

*Талецкий А.И.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Калугина М.А. – канд. физ-мат. наук*

Одно из наиболее популярных направлений в современном машинном обучении – методы классификации тональности различных данных, таких как фотографии лиц, тексты постов в интернете или записи речевых сигналов человека. Оценка тональности речевых сигналов отстает от других направлений и, в связи с этим, представляет особый интерес для различных исследований.

В работе используется датасет, загруженный с сайта стэнфордского университета и распространяемый по свободной некоммерческой лицензии. Датасет состоит из коротких аудиофайлов, каждый по 4 секунды. 12 мужчин и 12 женщин зачитывают 2 фразы в 8 тональных классах:

- нейтральность;
- спокойствие;
- счастье;
- грусть, печаль;
- злость, ненависть;
- испуг;
- удивление.

В результате варьирования различных параметров, не влияющих на общую тональность записи, получается датасет содержащий 2800 примеров.

Для оценки качества работы алгоритмов используется кросс-валидационная оценка [1]. Подсчет производится по следующим метрикам:

- precision;
- recall;
- f1-мера.

Результатирующим считалось значение f1-меры.

Для упрощения разработки и последующего расширения программного средства было принято решение строить его по принципу слабосвязанных модулей на основе принципов проектирования SOLID. А именно, программное средство было разделено на три модуля:

- модуль извлечения признаков;
- модуль обучения модели;
- модуль использования модели.

Модуль извлечения признаков извлекает признаки из тонально размеченного набора аудио файлов и выполняет предварительную обработку, стандартизируя извлеченные признаки. В качестве извлекаемых из аудио сигналов признаков выбраны мел-кепстральные коэффициенты, получаемые в результате двойного преобразования Фурье. В статье [2] обосновывается выбор мел-кепстральных коэффициентов в качестве признаков.

Задача модуля обучения модели состоит в обучении выбранной на текущей итерации модели. Обучается модель на основе подготовленных на предыдущем шаге и, далее, сериализуется для последующей оценки качества её работы.

Модуль использования модели поочередно загружает обученные модели и выполняет кросс-валидационную оценку их качества работы.

В результате получаются следующие оценки качества работы алгоритмов:

- наивный Байес: 0.2 (или 20%);
- метод опорных векторов 0.3 (или 30%)
- линейная регрессия 0.4 (или 40%)
- решающие деревья 0.56 (или 56%)
- ближайшие соседи 0.67 (или 67%)
- композиция градиентного бустинга и ближайших соседей 0.71 (или 71%)

В дополнение стоит сказать, что весьма эффективными для решения этой задачи могут оказаться рекуррентные нейронные сети и это может быть темой отдельного исследования.

**Список использованных источников:**

1. A Study of CrossValidation and Bootstrap for Accuracy Estimation and Model Selection, Ron Kohavi, Computer Science Department, Stanford University, Stanford, CA, 9405.
2. D. Wu, T. D. Parsons, S.S. Narayanan, "Acoustic feature analysis in speech emotion primitives estimation", 2010.

## **ОЦЕНКА ЭФФЕКТИВНОСТИ ПРИМЕНЕНИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ АНАЛИЗА ТОНАЛЬНОСТИ РЕЧЕВЫХ СИГНАЛОВ ЧЕЛОВЕКА**

*Талецкий А.И.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Калугина М.А. – канд. физ-мат. наук*

Одно из наиболее популярных направлений в современном машинном обучении – методы классификации тональности различных данных, таких как фотографии лиц, тексты постов в интернете или записи речевых сигналов человека. Оценка тональности речевых сигналов отстает от других направлений и, в связи с этим, представляет особый интерес для различных исследований.

Данная работа является логическим продолжением и развитием предыдущей. Где для решения задачи тестировались методы классического машинного обучения. Лучший результат среди классических методов показала композиция градиентного бустинга и ближайших соседей 0.71 (или 71%).

В работе используется датасет, загруженный с сайта стэнфордского университета и распространяемый по свободной некоммерческой лицензии. Датасет состоит из коротких аудиофайлов, каждый по 4 секунды. 12 мужчин и 12 женщин зачитывают 2 фразы в 8 тональных классах:

- нейтральность;
- спокойствие;
- счастье;
- грусть, печаль;
- злость, ненависть;
- испуг;
- удивление.

В результате варьирования различных параметров, не влияющих на общую тональность записи, получается датасет содержащий 2800 примеров.

Для оценки качества работы алгоритмов используется оценка на отложенных данных. Для тестирования стратифицировано по тональным классам откладывается 20% или 560 примеров. Подсчет производится по следующим метрике ассигасу.

Нейронная сеть состоит из слоев, собранных в блоки в следующем порядке:

- AveragePolling1D;
- GRU [1];
- Dense and Dropout.

Сеть имеет порядка 300 тысяч параметров. В качестве оптимизатора используется Adam [2]. Одна эпоха обучения с батчем в 64 объекта занимает 45 секунд. Уже к 10 эпохе сеть преодолевает бейзлайн в 71% и достигает максимума в 97% примерно к 25 эпохе. Эффект переобучения практически отсутствует. На тестовых данных сеть показывает результат в 0.8857 или примерно 89%.

Как и предполагалось рекуррентные нейронные сети оказались весьма эффективными для решения этой задачи. Прирост качества составил почти 20% что превышает все возможные ожидания.

В дополнение стоит отметить, что дополнительных улучшений можно достигнуть путем использования слоя Attention. Также имеет смысл попробовать одномерные сверточные сети.

**Список использованных источников:**

1.Cho, Kyunghyun; van Merriënboer, Bart; Gulcehre, Caglar; Bahdanau, Dzmitry; Bougares, Fethi; Schwenk, Holger; Bengio, Yoshua (2014). "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation".

2.Diederik P. Kingma; Jimmy Ba; (2014). "Adam: A Method for Stochastic Optimization".

## **ПРИЛОЖЕНИЕ ДЛЯ КОНТРОЛЯ ПРОИЗВОДИТЕЛЬНОСТИ РАБОЧИХ ГРУПП**

*Туманов Д.Е.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Анисимов В.Я. – к.ф.-м.н., доцент*

Цель работы – разработка программного продукта, автоматизирующий операции с данными о работе сотрудников предприятия, их обработка, консолидация и редактирование. Использование данного программного продукта позволит упростить отслеживание изменений в рабочем процессе различных групп сотрудников предприятия, а также предугадать данные изменения заранее. Уменьшается количество т.н. «бумажной» работы, поскольку все операции автоматизированы, уменьшается вероятность ошибки благодаря протестированным системам подсчета.

В настоящее время существует множество компаний с налаженными и протестированными временем процессами по управлению различными проектами, работой на предприятии и т.д. Но при увеличении объема задач, их общего количества и количества сотрудников, привлекаемых на данные задачи, отслеживание хода работы становится трудной задачей, не говоря уже о планировании новых задач и анализе производительности.

Поэтому сейчас всё больше компаний различной направленности начинают внедрять в свои процессы компоненты, автоматизирующие управление проектами, командами и отслеживающие изменения и тренды в производительности труда. Данные приложения позволяют пользователям в режиме реального времени управлять проектами, различными задачами и активностями сотрудников, контролировать рабочие процессы на предприятиях, осуществлять анализ проделанной работы и вносить корректировки в планирование будущих задач.

Решение о создании приложения для контроля производительности рабочих групп было намеренным, и вызвано оно тем, что информация о производительности работника при решении различных задач, его занятости и успешности выполнения поставленных работ важна не только руководителю предприятия или группы рабочих, но также и самому работнику. Разработанное приложение позволяет получить статистику о проводимых работах на предприятии, позволяет отслеживать различного рода отклонения в работе различных юнитов на предприятии (целых проектов или же отдельных менеджеров и сотрудников), сравнивая характеристики данных юнитов, полученных ранее. Только на основе данных о работе отдельных сотрудников можно уже спрогнозировать тенденцию развития различных проектов на предприятии в ближайшем будущем.

Приложение обеспечивает автоматизацию, ускоряет и упрощает работу, сводя тем самым весь труд некоторых отделов к обычному заполнению отчетов. Успешное развитие современного бизнеса во многом зависит от широкого применения новейших информационных технологий, позволяющих обрабатывать информацию любого вида с наибольшей эффективностью. На рисунке 1 изображена Схема интеграции приложения в сеть предприятия.

Неотъемлемой частью данного приложения является понятие сервисов Active Directory, служба каталогов корпорации Microsoft для операционных систем семейства Windows Server, позволяющей администраторам использовать групповые политики для обеспечения единообразия настройки пользовательской рабочей среды, разворачивать программное обеспечение на множестве компьютеров через групповые политики или посредством System Center Configuration Manager,

устанавливать обновления операционной системы, прикладного и серверного программного обеспечения на всех компьютерах в сети, используя Службу обновления Windows Server.

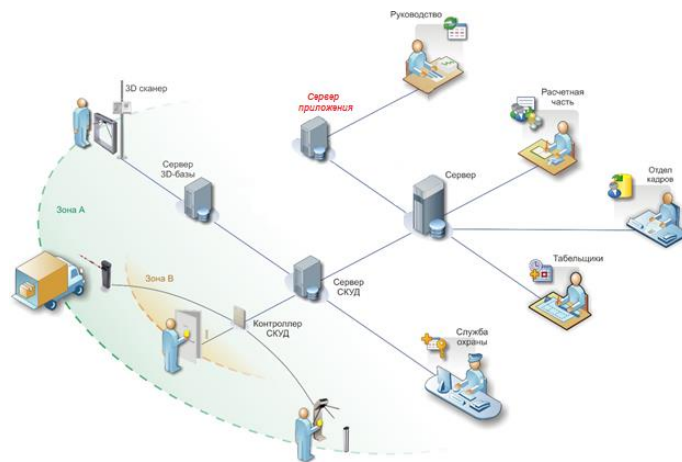


Рисунок 1 – Схема интеграции приложения в сеть предприятия

Данное приложение может предоставлять пользователю статистику о работе предприятия за текущий месяц и год. Данная статистика может включать в себя информацию о часах работы в текущем месяце (сколько было проработано и оплачено, сколько часов ожидалось и т.д.), часах работы за текущий год (представление в виде графика), часах, потраченных на различные типы проектов, и таблицу, которая включает в себя более подробные данные, сгруппированные по работникам, проектам и менеджерам предприятия. На рисунке 2 изображена общая схема работы приложения.

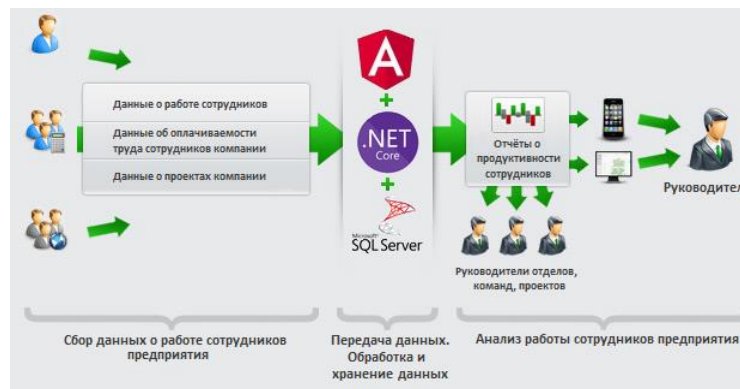


Рисунок 2 – Общая схема работы приложения

**Список использованных источников:**

1. Freeman A. Pro ASP.NET Core MVC – 2017. – P. 354-371.
2. Fowler M. Patterns of Enterprise Application Architecture – 2002. – P. 202-223.
3. Fritchey G. SQL Server Query Performance Tuning. – 2016. – Vol. 4 – P. 155-168,
4. Ullman J. D., Widom J. A First Course in Database Systems. – 2014. – P. 232-240.

## СИСТЕМА ПОДБОРА РАЗРАБОТЧИКОВ ДЛЯ ВЫПОЛНЕНИЯ ЗАДАЧ НА КРУПНЫХ ПРОЕКТАХ

Туманов Д.Е.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Анисимов В.Я. – к.ф.-м.н., доцент

В докладе рассматриваются вопросы создания системы подбора разработчиков для выполнения задач на крупных проектах, способ обработки текстов задач.

Интерес к профессиям, связанным с информационными технологиями, в последние годы заметно вырос — это видно, например, по тому, как увеличилась доля запросов в поисковых системах по таким ключевым словам как разработчик, разработка, программист или программирование. В 2019 году она была на четверть больше, чем в 2017. Спрос на ИТ-специалистов растёт, за два года доля объявлений о работе для них от всех объявлений на HeadHunter увеличилась на 5,5%. С ростом интереса и количества разработчиков растут и ИТ-компании. За 2019 год белорусский офис EPAM нанял 1600 технических специалистов. IBA Group – 200 технических сотрудников, данный показатель чуть выше, на 100 человек, чем у компании Wargaming. Лидером в процентном приросте штата стала Synesis: по сравнению с 2018 годом — +210%, с 35-й на 11-ую строчку рейтинга сайта dev.by. Но с ростом компаний и их технического штата растут также и затраты на управление, анализ работы, подбор команд из специалистов различных технических направлений. Для этого и было создано рассматриваемое приложение, цель которого – минимизировать расходы на подбор технического специалиста для решения проектных задач.

В основе приложения, а именно в обработке задач, лежат идеи обработки естественного языка и глубоких нейронных сетей.

Направление обработки естественного языка на данный момент является одним из важнейших в исследованиях в области искусственного интеллекта. Это направление зародилось с появлением первых вычислительных машин и с идеей об их использовании для решения задач, связанных с естественным языком, на котором разговаривают и пишут люди. Одной из первых стояла задача машинного перевода – автоматического перевода текста с одного языка на другой с помощью компьютера или вычислительной машины. Для пары «русский-английский» были выполнены первые успешные эксперименты по машинному переводу еще в середине 1950-х годов. Вторая задача – создание диалоговых систем, способных вести с человеком диалог на естественном языке. В те же годы появились первые подобные программы и системы. Третья задача – создание вопросно-ответных систем, способных точно отвечать на вопрос, заданные человеком. Вопросы при этом обычно представляются в форме текста на естественном языке. В последнее время круг задач по обработке естественного языка стал намного шире.

Для обработки текста задачи и подбора разработчиков по данному критерию было принято решение использовать глубокие нейронные сети с рекуррентными и сверточными слоями.

Рекуррентная нейронная сеть представляет отдельный вид нейронных сетей, где связи между элементами образуют направленную последовательность, благодаря чему появляется возможность обрабатывать серии событий во времени или последовательные пространственные цепочки. Главное отличие данного вида нейронных сетей от многослойных перцептронов – рекуррентные сети могут использовать свою внутреннюю память для обработки последовательностей произвольной длины. Это позволяет применять данные сети в таких задачах, как распознавание рукописного текста или распознавание речи, где целостная структура входных данных разбита на части. Применяя сверточные слои в качестве начальных слоев в рекуррентных нейронных сетях при обработке текста, можно добиться достаточного представления входных данных, встраивая его в вектор фиксированной длины, который можно использовать для дальнейших задач. Сегодня всё чаще можно увидеть повсеместное использование сверточных и рекуррентных слоёв в рамках одной нейронной сети.

Пользователь может ввести информацию о задаче несколькими способами:

- ввод текста задачи;
- ввод ссылки на сторонний ресурс с текстом задачи (в данный момент поддерживается только ресурсы из Jira – распространенный инструмент управления проектами);

При обучении модели нейронной сети был достигнут результат в 91%. При попытке использовать для подбора разработчика на основе текста задачи более простые инструменты, например многоклассовую логистическую регрессию с предварительной векторизацией текстов, максимальный из полученных результатов составил 65%, поэтому и было принято решение об использовании глубоких нейронных сетей.

Последующими шагами в развитии системы будет улучшения качества модели, а также написание дополнительных инструментов для более удобного способа ввода задачи в систему (например браузерные расширения и т.д.).

**Список использованных источников:**

1. Rogerson J. Theory, Concepts and Methods of Recurrent Neural Networks and Soft Computing // Clanrye International – 2015. – P. 254-270.
2. Hasim S. Long Short-Term Memory recurrent neural network architectures for large scale acoustic modeling – 2014. – P. 2–3.
3. Graves A., Liwicki M., Fernandez S., Bertolami R., Bunke H., Schmidhuber J. A novel connectionist system for unconstrained handwriting recognition // IEEE Transactions – 2009. – Vol. 31 – P. 855–868,
4. Dean J., Corrado G., Monga R., Chen K., Devin M., Le Q. V., Mao M. Z., Ranzato M., Senior A. W., Tucker P. A., Yang K., Ng A. Y. Large scale distributed deep networks. // NIPS – 2012. – P. 1232–1240.



# ИНТЕГРИРОВАНИЕ ВЫРАЖЕНИЙ СОДЕРЖАЩИХ ПРОИЗВЕДЕНИЯ ПОКАЗАТЕЛЬНЫХ И АЛГЕБРАИЧЕСКИХ ФУНКЦИЙ

Христенко А. В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Теслюк В.Н. – доцент, канд. физ.-мат. наук

В данной работе рассмотрены основные виды выражений, содержащих произведения показательных и алгебраических функций, вычисляемые в конечном виде. Представлен вывод формул для их расчета.

1) Применим формулу интегрирования по частям:

$$\int x^m e^{nx} dx = \frac{1}{n} x^m e^{nx} - \frac{m}{n} \int x^{m-1} e^{nx} dx.$$

Продолжая последовательно применять это формулу и сделав несложные преобразования получим:

$$\int x^m e^{nx} dx = \frac{e^{nx}}{n^{m+1}} \sum_{p=0}^{p=m} (-1)^p \binom{m}{p} p! n^{m-p} x^{m-p} \quad (1).$$

Интегрируем по частям функцию другого вида:

$$\int \frac{e^{nx}}{x^m} dx = -\frac{e^{nx}}{(m-1)x^{m-1}} + \frac{n}{m-1} \int \frac{e^{nx}}{x^{m-1}} dx.$$

Последовательно применяя эту формулу и выполняя некоторые преобразования получаем выражение:

$$\int \frac{e^{nx}}{x^m} dx = -\frac{e^{nx}}{(m-1)x^{m-1}} \sum_{p=0}^{p=m-2} \frac{n^p x^p}{\binom{m-2}{p} p!} + \frac{n^{m-1}}{(m-1)!} \int \frac{e^{nx}}{x} dx \quad (m \geq 2) \quad (2).$$

Интеграл  $\int \frac{e^{nx}}{x} dx$  в (2) не выражается в конечном виде и выражается бесконечным рядом. По формуле Маклорена:

$$\int \frac{e^{nx}}{x} dx = \ln x + \frac{nx}{1 \cdot 1!} + \frac{n^2 x^2}{2 \cdot 2!} + \frac{n^3 x^3}{3 \cdot 3!} + \dots$$

Получим виды формул (1) и (2) для отрицательно  $n$ :

$$\int x^m e^{nx} dx = -\frac{e^{nx}}{n^{m+1}} \sum_{p=0}^{p=m} \binom{m}{p} p! n^{m-p} x^{m-p} \quad (3);$$

$$\int \frac{e^{nx}}{x^m} dx = -\frac{1}{(m-1)x^{m-1} e^{nx}} \sum_{p=0}^{p=m-2} \frac{(-1)^p n^p x^p}{\binom{m-2}{p} p!} + (-1)^p \frac{n^{m-1}}{(m-1)!} \int \frac{1}{x e^{nx}} dx \quad (m \geq 2) \quad (4).$$

Интеграл  $\int \frac{1}{x e^{nx}}$  не выражается в конечном виде.

При помощи равенства  $a^{nx} = e^{nx \ln a}$  интегралы  $\int x^m a^{nx} dx$  и  $\int \frac{a^{nx}}{x^m} dx$  приводятся к виду  $\int x^m e^{nx \ln a} dx$  и  $\int \frac{e^{nx \ln a}}{x^m} dx$ . Выражая последние интегралы по формулам (1) и (2) получим:

$$\int x^m a^{nx} dx = \frac{a^{nx}}{(n \ln a)^{m+1}} \sum_{p=0}^{p=m} (-1)^p \binom{m}{p} p! (x n \ln a)^{m-p} \quad (5);$$

$$\int \frac{a^{nx}}{x^m} dx = -\frac{a^{nx}}{(m-1)x^{m-1}} \sum_{p=0}^{p=m-2} \frac{(x n \ln a)^p}{\binom{m-2}{p} p!} + \frac{(n \ln a)^{m-1}}{(m-1)!} \int \frac{a^x}{x} dx \quad (m \geq 2) \quad (6).$$

В этих формулах  $n$  может быть отрицателен.

2) Исследуем интеграл  $\int f(x) e^{nx} dx$ , где  $f(x) = a_0 x^m + a_1 x^{m-1} + \dots + a_m$  – целая рациональная функция степени  $m$ , при последовательном применении формулы интегрирования по частям

получим:

$$\int f(x)e^{nx} = \frac{1}{n}f(x)e^{nx} - \frac{1}{n^2}f'(x)e^{nx} + \dots + \frac{(-1)^m}{n^{m+1}}f^{(m)}(x)e^{nx} = F(x)e^{nx} \quad (7),$$

где  $F(x) = A_0x^m + A_1x^{m-1} + \dots + A_m$  – также целая рациональная функция степени  $m$ . Коэффициенты  $A_0, \dots, A_m$  можно найти методом неопределенных коэффициентов.

3) Интеграл  $\int x^{2m+1}e^{nx^2} dx$  подстановкой  $x^2 = y$  приводится к интегралу  $\frac{1}{2} \int y^m e^{ny} dy$ . Применяя к последнему формулу (1) получится:

$$\int x^{2m+1}e^{nx^2} dx = \frac{e^{nx^2}}{2n^{m+1}} \sum_0^m (-1)^p \binom{m}{p} p! n^{m-p} x^{2(m-p)} \quad (8).$$

Аналогично получаем:

$$\int x^{2m+1}a^{nx^2} dx = \frac{a^{nx^2}}{2(n \ln a)^{m+1}} \sum_0^m (-1)^p \binom{m}{p} p! (n \ln a)^{m-p} x^{2(m-p)} \quad (9).$$

Заменяя в последних двух формулах  $n$  на  $-n$  окончательно получим следующее:

$$\int x^{2m+1}e^{-nx^2} dx = -\frac{e^{-nx^2}}{2n^{m+1}} \sum_0^m \binom{m}{p} p! n^{m-p} x^{2(m-p)} \quad (10);$$

$$\int x^{2m+1}a^{-nx^2} dx = \frac{a^{-nx^2}}{2(n \ln a)^{m+1}} \sum_0^m \binom{m}{p} p! (n \ln a)^{m-p} x^{2(m-p)} \quad (11).$$

Интегралы  $\int x^{2m}e^{\pm nx^2} dx$  и  $\int x^{2m}a^{\pm nx^2} dx$  не вычисляются в конечном виде.

**Список использованных источников:**

1. Тимофеев, А.Ф. Интегрирование функций: учеб. пособие – М.: Изд-во Техн.-Теор. Лит-ры, 1948. – 348 с.
2. Жевняк, Р.М. Высшая математика. Ч. 1 / Р. М. Жевняк, А. А. Карпук. – Минск: Выш. шк., 1992. – 359 с.
3. Жевняк, Р.М. Высшая математика. Ч. 2 / Р. М. Жевняк, А. А. Карпук. – Минск: Выш. шк., 1993. – 108 с.

## УЯЗВИМОСТЬ FLASH NAND

*Цыбулько К.Д.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Заливако С.С. – канд. тех. наук*

В статье освещается проблема хранения информации на NAND-флэш носителях. Рассматриваются проблемы, связанные с возникновением различного рода ошибок и уязвимостей, а также пути их сглаживания и устранения.

За последние годы накопители информации на основе NAND-флэш памяти (структурная схема приведена на рис. 1) стали доминирующими в электронных устройствах различного назначения (USB-флэш накопители, карты памяти различных типов, мобильные устройства, такие, как телефоны, фотоаппараты, медиаплееры и др.). NAND-флэш - это форма энергонезависимой памяти (Electrically Erasable Programmable Read-Only Memory, EEPROM), которая может быть электрически стерта и записана. Накопитель состоит из массива элементов флэш-памяти и контроллера, который обрабатывает запросы на чтение или запись данных и, таким образом, представляет собой интерфейс между флэш-памятью и хостом. Основными причинами широкого распространения NAND-флэш является то, что ее емкость постоянно растет, а стоимость (количество бит на цент) уменьшается. Такой эффект является результатом двух тенденций: эффективного масштабирования технологических процессов и многоуровневого кодирования ячеек (multi-level cell, MLC; triple-level cell, TLC; qual-level cell, QLC) [1]. С другой стороны, становится все труднее обеспечивать надежность хранения данных во флэш-памяти. Указанные выше тенденции приводят к уменьшению количества электронов в транзисторе с плавающим затвором и усилению взаимного влияния элементов памяти.

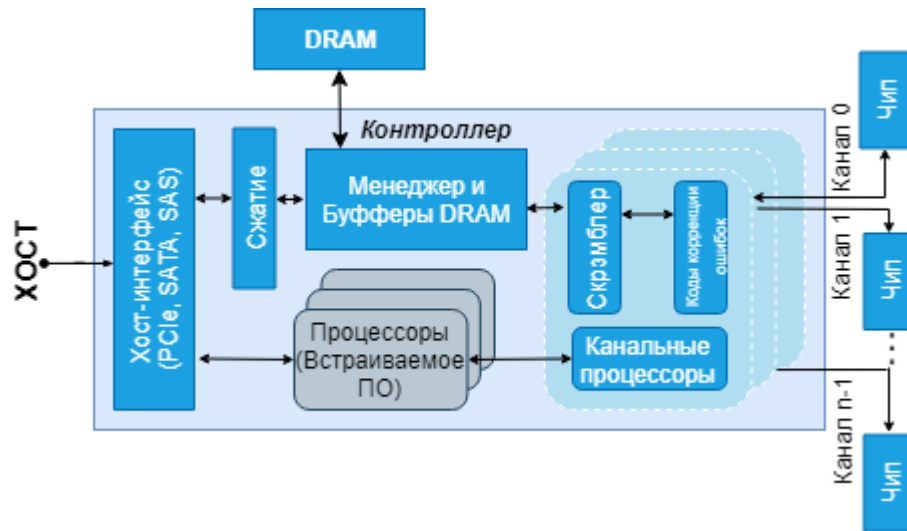


Рис.1 – Структурная схема NAND-флэш накопителя

Возникновение ошибок во флэш-памяти во многом зависит от данных, хранящихся в ячейках, и от работы различных приложений. Для уменьшения зависимости частоты ошибок от данных и предотвращения записи данных вредоносными приложениями, контроллер скремблирует (scrambling) данные перед записью. Основная концепция скремблирования заключается в генерации псевдослучайной последовательности нулей и единиц, для того чтобы минимизировать любые ошибки, которые зависят от данных. Реализация скремблирования приведена на рис. 2 и выполняется с использованием линейного сдвигового регистра с обратной связью (Linear Feedback Shift Register, LFSR). Так как скремблирование и дескремблирование с помощью LFSR реализуются одинаково (за счет свойств операции XOR), для корректного выполнения дескремблирования необходимо только знание начального состояния регистра [2]. В связи с этим для каждой записываемой страницы данных в качестве начального состояния LFSR может использоваться логический адрес этой страницы (таким образом, страница может быть правильно дескремблирована, даже если операции обслуживания переносят страницу в другое физическое местоположение). Однако, стоит отметить, что вредоносное приложение может обойти скремблирование. Злоумышленник воссоздает скремблер в программном обеспечении, которое имитирует его работу. Поскольку все необходимые составляющие для скремблирования известны (логический адрес страницы и неприводимый полином для LFSR), дескремблированные данные из вредоносной программы могут быть записаны в SSD.



Рис.2 – Структурная схема скремблера

Помимо вредоносного программного обеспечения флэш-память подвержена случайным ошибкам. Есть два основных источника таких ошибок. Они возникают в частично запрограммированных ячейках памяти. Первым источником ошибок является результат взаимного влияния программ, приводящее к тому, что пороговое напряжение ячейки увеличивается при программировании соседней. Второй тип ошибок возникает при чтении. Оно приводит к существенному сдвигу порогового напряжения для ячейки с более низким напряжением. Таким образом, незапрограммированные и частично запрограммированные ячейки будут сильнее подвержены такого рода ошибкам, так как они имеют более низкие пороговые напряжения.

Ошибки могут быть использованы вредоносными программами для повреждения данных. Есть

две модели использования ошибок: на основе межпрограммной интерференции между ячейками и возникновении ошибок при чтении [3]. Идея модели, основанной на межпрограммной интерференции, состоит в том, что вредоносное приложение может вызывать множество ошибок на странице другого приложения, тем самым разрушая страницу и сокращая срок службы SSD. Целью первой модели является запись вредоносным приложением наихудшей комбинации данных (например, все нули) на страницу таким образом, чтобы вызвать максимальное количество программных ошибок. Модель нарушения чтения использует тот факт, что двухэтапное программирование значительно повышает уязвимость как частично запрограммированных, так и незапрограммированных элементов флэш-памяти. Механизм работы второй модели заключается в том, что вредоносное приложение быстро выполняет большое количество операций чтения за очень короткий промежуток времени, тем самым, вызывая ошибки при чтении, которые повреждают частично запрограммированные страницы. В данной модели вредоносному приложению не нужно иметь дополнительной информации об атакуемом приложении. Так же, программа может атаковать сразу несколько страниц флэш-памяти, и повреждать страницы, записанные после совершения атаки.

В настоящее время применяются следующие решения для устранения описанных выше атак: буферизация данных DRAM в контроллере, адаптивное считывание опорного напряжения и использование нескольких проходных напряжений [4]. Последние два решения устраняют от 20% до 70% возникающих ошибок и могут быть использованы не для всех типов атак. Самым эффективным решением, согласно [4], является буферизация данных в контроллере, которая полностью устраняет возникающие ошибки при чтении данных и межпрограммной интерференции. Однако вызывает задержку от 1% до 16% при чтении данных и требует дополнительной памяти до 2 Мб. Таким образом, издержки на исправления значительно увеличиваются. Для повышения надежности NAND-флэш памяти необходимо пересмотреть существующие решения, в том числе связанные с равномерной записью информации и добавлением к контроллеру детектора напряжения.

Равномерная запись информации позволит уменьшить изнашиваемость с помощью перераспределения информации по всем элементам памяти, увеличивая тем самым общий срок службы устройства. Детектор напряжения будет блокировать операции с флэш-памятью при выходе питающего напряжения за допустимые границы, тем самым позволяя избежать ошибок связанных с взаимодействием между элементами памяти. Данный метод будет почти со 100% вероятностью устранять ошибки связанные с взаимодействием между элементами памяти, однако метод будет требовать дополнительный аппаратуры и, соответственно, стоимость накопителя повысится.

**Список использованных источников:**

1. Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, Onur Mutlu/ Error Characterization, Mitigation and Recovery in Flash-Memory-Based Solid-State Drives (Proceedings of the IEEE, September 2017);
2. J.P. van Zandwijk /A mathematical approach to NAND false-memory descrambling and decoding /Digital Investigation 12 (2015) 41e5242
3. Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, Onur Mutlu/ Vulnerabilities in MLC NAND Flash Memory Programming: Experimental Analysis, Exploits, and Mitigation Techniques;
4. Y. Cai et al., "Characterization and Mitigation of Reliability and Security Vulnerabilities in MLC NAND Flash Memory Programming," Carnegie Mellon Univ., SAFARI Research Group, Tech.Rep.2017-002, 2017.

## РЕКУРРЕНТНАЯ НЕЙРОННАЯ СЕТЬ

*Чибисов И.В., Клапатов И.А., Виноградов А.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Стержанов М.В. – кан. тех.наук, доцент*

Рекуррентная нейронная сеть представима в виде сети с прямым распространением сигнала. На вход подаются элементы последовательности. При поступлении элемента возвращается два значения, первое из которых идет на выход из сети, а второе поступает на вход копии нейронной сети. И так до последнего элемента, от него возвращается только одно значение: на выход. Таким образом рекуррентная нейронная сеть позволяет работать с последовательностями любой длины.

Рекуррентную нейронную сеть можно представить в виде сети с прямым распространением сигнала. В такой сети используется прием, который называется разворачивание во времени: создается несколько копий рекуррентной нейронной сети (рис. 1).

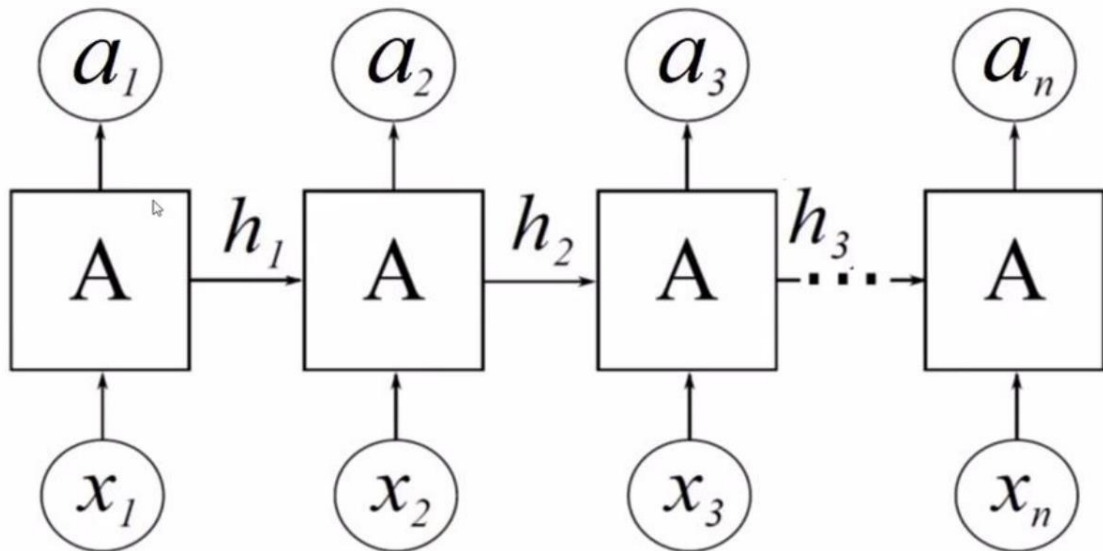


Рисунок 1. Разворачивание рекуррентной нейронной сети во времени

На вход нейронной сети поступают элементы последовательности. На вход первой копии поступает первый элемент последовательности  $x$ , следующий элемент поступает на вход второй и так далее, пока мы не дойдем до последней копии. Рекуррентная нейронная сеть возвращает два значения. Первое значение -  $a$ , поступает на выход из нейронной сети. Также рекуррентная нейронная сеть выдает второе значение -  $h$ , которое поступает на вход следующей копии нейронной сети. Это значение называют скрытым состоянием, оно учитывает то, что было на предыдущих этапах анализа последовательности. Следующая копия нейронной сети  $A$  на вход получает второй элемент последовательности  $x$ , а также скрытое состояние с предыдущего этапа. Вторая копия нейронной сети анализирует одновременно текущий элемент последовательности и данные со скрытого состояния предыдущей копии нейронной сети, и в зависимости от результатов анализа также выдает два значения: выходное значение  $a_2$  и скрытое состояние  $h_2$ , которое передается следующей копии нейронной сети. И так продолжается, пока мы не дойдем до последнего элемента данных в последовательности. Для него рекуррентная сеть выдает уже одно значение -  $a_n$ , которое подается на выход без скрытого состояния. В отличие от полносвязной нейронной сети рекуррентная нейронная сеть может работать с последовательностями входных данных любой длины. Для этого нужно создать столько копий рекуррентной нейронной сети, развернутой во времени, сколько элементов входных данных последовательности подается на вход.

Рекуррентные нейронные сети имеют широкое применение в задачах, где нечто целостное разбито на составные части, это может быть распознавание рукописного текста или речи. В таких задачах они особенно эффективны.

**Список использованных источников:**

1. Каллан, Роберт, Основные концепции нейронных сетей: Пер. с англ. - М.: Издательский дом "Вильямс", 2001.
2. Джанелидзе, Г.М. Рекуррентные нейронные сети // Рекуррентные нейронные сети в задаче анализа тональности текста: Выпускная квалификационная работа бакалавра СПбГУ. - СПб.2016.-С 13 - 14

## ИССЛЕДОВАНИЕ РАСПРОСТРАНЕНИЯ СИГНАЛОВ НА ПАРАМЕТРИЧЕСКОЙ МОДЕЛИ ФНФ ТИПА АРБИТР

Шамына А.Ю.

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Иванюк А.А. – д-р. техн. наук, профессор*

В настоящей работе рассмотрено построение параметрической модели физически неклонированной функции типа арбитра (АФНФ) средствами САПР Хіііпх 14.7. Описана постановка эксперимента по временной оценке асимметрии прохождения копий тестового сигнала на различных звеньях блока симметричных путей (БСП) АФНФ, а также произведен анализ полученных результатов.

Существующие реализации АФНФ базируются на уникальности трансляции тестовых сигналов через БСП для каждого изделия [1]. Уникальность распространения тестовых сигналов через БСП АФНФ обусловлена различными инерциальными задержками коммутационных звеньев и транспортными задержками между ними. Однако в большинстве работ оценка асимметрии тестовых сигналов производится на выходах последнего узла БСП либо на арбитраже, что затрудняет определение удельного вклада каждого звена БСП в конечный результат. Также остается открытым вопрос оценки доли вклада в общую энтропию инерциальных и транспортных задержек.

Для изучения АФНФ была создана параметрическая модель «Post place & route» с использованием САПР Xilinx 14.7 и HDL языка Verilog. Тестовые модули были описаны также на Verilog. Благодаря параметризации количества звеньев БСП созданная модель может быть применена для АФНФ различной размерности. Модель строилась для FPGA Xilinx Artix 7. При имплементации проекта была отключена логическая оптимизация, т.к. иначе после технологического синтеза каскад мультиплексоров БСП может быть отличным от ожидаемого.

Как правило, звено БСП имеет 3 входа (2 для тестовых сигналов и 1 для разряда запроса) и 2 выхода. В рамках реализации АФНФ на FPGA звено БСП состоит из двух lut-компонентов. Входы «a» и «b» являются входами тестовых сигналов, вход «c» представляет разряд запроса к АФНФ, а «x» и «y» являются выходами тестовых сигналов из звена БСП. На рисунке 1 представлен результат технологического синтеза звена БСП.

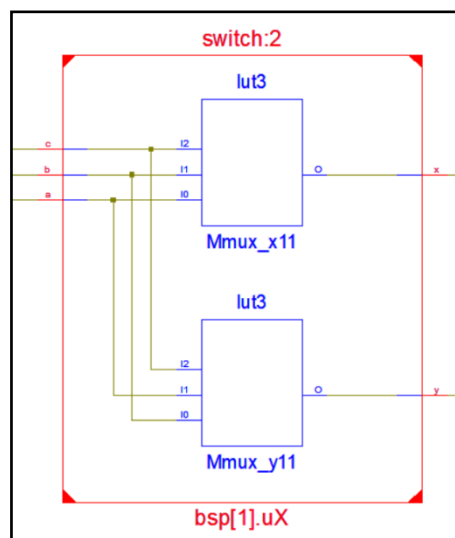


Рисунок 1 – технологический синтез звена БСП

В работе [2] модельные эксперименты проводились с учетом асимметрии входных сигналов на входах первого звена БСП. Использование языка Verilog при описании тестовых модулей позволило устранить эту асимметрию благодаря возможности установки значений внутренних сигналов тестируемого модуля напрямую. Также использование Verilog открыло возможность оценки временной асимметрии фронтов тестовых сигналов на выходах каждого звена БСП. Эта возможность позволяет за один запуск модельного эксперимента оценить временные разницы между фронтами при различном количестве звеньев БСП.

На временной диаграмме сигналов звена БСП (рисунок 2) можно проследить характерную асимметрию фронтов тестовых сигналов на выходах из звена, несмотря на полную симметричность сигналов на входах. Эта разница обусловлена внутренней асимметричностью путей сигналов, а также инерционностью коммутационных блоков. Для выбранного звена асимметричность на выходах составила 407 пс при прямой трансляции сигналов внутри блока и 63 пс при перекрестной соответственно.

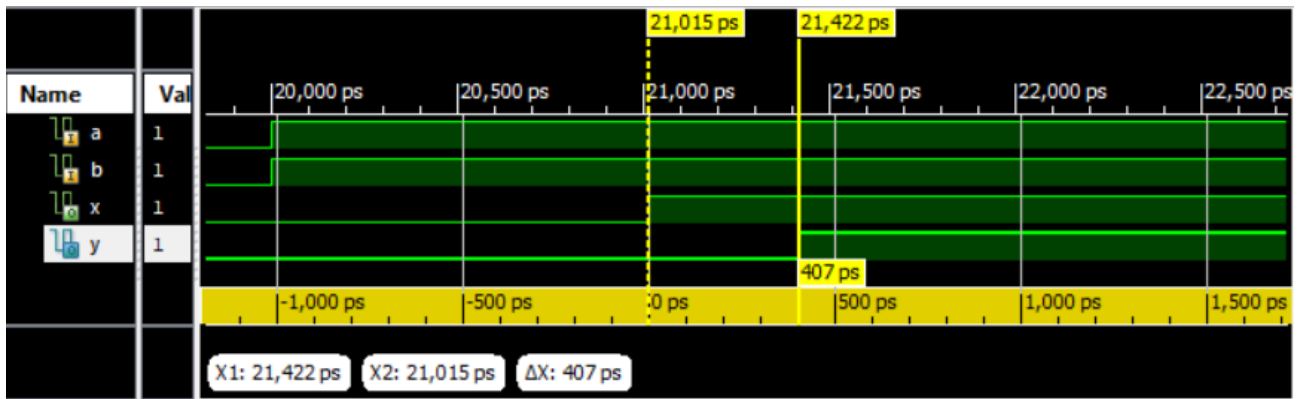


Рисунок 2 – временная диаграмма сигналов звена БСП

При проведении модельного эксперимента были сгенерированы все возможные запросы при разрядности 16. Для каждого запроса оценена временная асимметричность прохождения фронтов сигналов на выходах последнего звена БСП. Для анализа плотности временного распределения оценок асимметричности диапазон полученных значений был разбит на интервалы и осуществлен подсчет вхождений результатов в эти интервалы. Полученный результат представлен в виде гистограммы на рисунке 3

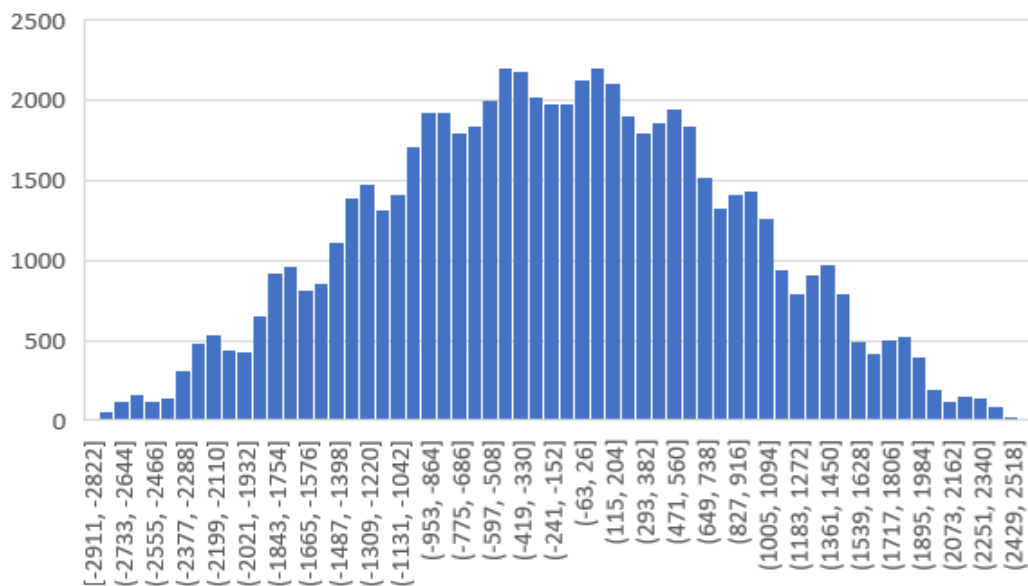


Рисунок 3 – гистограмма временных разниц фронтов сигнала на различных интервалах

Из рисунка 3 видно, что большинство результатов сосредоточено в потенциально метастабильном регионе (когда временная разница между фронтами тестовых сигналов близка к 0). Это может негативно сказаться на характеристиках ФНФ и потребовать применения дополнительных решений.

**Список использованных источников:**

1. Заливако, С. С. Физически неклонированные функции / Заливако С. С., Иванюк А. А. // Информационные технологии и системы 2019 (ИТС 2019) = Information Technologies and Systems 2019 (ITS 2019) : материалы международной научной конференции, Минск, 30 октября 2019 г. / Белорусский государственный университет информатики и радиоэлектроники; редкол. : Л. Ю. Шилин [и др.]. – Минск, 2019. – С. 8 – 21.
2. Шамына, А. Ю. Анализ симметричных путей физически неклонированной функции на FPGA / Шамына А. Ю., Иванюк А. А. // Информационные технологии и системы 2019 (ИТС 2019) = Information Technologies and Systems 2019 (ITS 2019) : материалы международной научной конференции, Минск, 30 октября 2019 г. / Белорусский государственный университет информатики и радиоэлектроники; редкол. : Л. Ю. Шилин [и др.]. – Минск, 2019. – С. 150 – 151.

**СЕКЦИЯ «ФИЛОСОФИЯ, ЛОГИКА И ЖИЗНЬ»**



## ФИЛОСОФСКИЕ И ГУМАНИСТИЧЕСКИЕ АСПЕКТЫ МЕЦЕНАТСТВА КАК СОЦИАЛЬНОГО ЯВЛЕНИЯ

*Шестакова А.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Шепетюк В.В. – кан. фил. наук*

Тема научной работы посвящена исследованию мировоззренческих основ меценатства, как социокультурного явления в контексте исторической ретроспективы белорусского государства. Рассматриваются актуальные вопросы мотивов и целей деятельности меценатов, благотворительных и волонтерских организаций как проводников гуманистических ценностей и хранителей потенциала духовной культуры общества. Анализируется содержание понятий «благотворительность», «меценатство», «спонсорство», а также различные формы их воплощения в реальной жизни. Делается попытка обобщения материала с целью выявления возможностей расширения участия широких слоев населения в эффективной благотворительной деятельности в современных социокультурных условиях. Для подготовки материалов использовались как исторические источники (документы Белорусского государственного архива-музея литературы и искусства, епархиальные ведомости Национального Исторического Архива Беларуси), так и результаты опросов, интервью с представителями меценатства и волонтерского движения.

Меценатство, благотворительность, спонсорство – понятия очень похожие, по смыслу тесно связанные между собой. Благодетель, жертвовател, покровитель, филантроп – эти слова употреблялись в истории для характеристики лиц, оказывающих безвозмездную помощь тем, кто в ней нуждался. В современном мире с его ускоренным ритмом жизни эти воззрения часто забываются либо подвергаются сомнению. Вместе с тем люди, как и прежде, по своей природе все также нуждаются в соприкосновении с примерами альтруизма, бескорыстия, великодушия, позволяющими им стать духовно сильнее и получить жизнеутверждающее ощущение важности своего труда и помощи другим людям, будь то материальная, денежная поддержка малоимущим или помощь и содействие миру искусства, науки, культуры. К побудительным причинам благотворительности, таким образом, относится осознание ее участниками ценностей взаимной поддержки и помощи, целостности и взаимозависимости различных представителей человеческого общества.

На актуальность темы работы указывает тот факт, что долгое время о благотворительности и меценатстве в новейшей истории Беларуси не было широко известно, но в настоящее время произошло множество позитивных изменений. Вначале робко о себе, как о меценатах, заявили некоторые крупные фирмы, компании, со временем их число увеличилось, расширился спектр оказываемой ими помощи, преумножилось число благотворительных инициатив. За последнее десятилетие в республике появились различные благотворительные фонды, получило развитие волонтерское движение и культурные межнациональные проекты. Прослеживается преемственность традиций взаимопомощи, заложенных еще нашими предками путем развития народной благотворительности и оказания поддержки, например, через сферу образования с целью формирования у населения нового образа жизни и мышления, чему немало способствовала, в частности, деятельность священнослужителей еще в начале XX в. (деятельность Г. Шимановского и П. Трусовского в деревнях Родковщина и Дукора), когда белорусские земли входили в состав Российской Империи [1].

Используя данные исторические аналогии, нами была поставлена цель выявить основные тенденции развития меценатства и благотворительности на территории Беларуси на примере деятельности Григория Шимановского и Петра Трусовского. Для достижения цели необходимо было выполнить следующие задачи: определить основные формы благотворительности и меценатства на территории Беларуси в период Великого княжества Литовского (далее – ВКЛ) и Речи Посполитой (далее – РП), выявить изменения в развитии благотворительности и меценатства на территории Беларуси в период Российской империи, охарактеризовать деятельность священников Г. Шимановского и П. Трусовского как меценатов и благотворителей, а также проследить основные тенденции развития меценатства и благотворительности на территории современной Беларуси.

Следует отметить, что в настоящее время под меценатом понимают лицо (юридическое или физическое), оказывающее на добровольной и безвозмездной основе материальную помощь из личных средств на развитие культуры и искусства, сферы образования, охраны культурного наследия, науки и научных исследований, спорта [2]. История данного понятия корнями уходит в глубокую древность: около 70 г. до н.э. при древнеримском императоре Октавиане Августе доверенным лицом был государственный деятель и покровитель искусств Гай Цильний Меценат. Его имя, как поклонника изящных искусств и покровителя поэтов, сделалось нарицательным. Благодаря Меценату, при поддержке императора, значительная часть финансов Римской империи направлялась на поощрение и развитие культуры, науки и мира искусств, воспитание духовности в обществе.

Под благотворительностью, в свою очередь, подразумевают пожертвование на какие-то определенные цели, как денежных средств, так и материальных составляющих: вещей, медикаментов, продуктов, любых иных ресурсов. Некоторые ученые-аналитики (А. Марченко, Я.

Буздуган, С. Долгий) к формам благотворительности относят гуманитарную помощь, миротворческую благотворительную практику, волонтерство, донорство [3]. При этом помощь может быть направлена на поддержку науки, образования, здравоохранения и в качестве социального содействия конкретным учреждениям: детским домам, обществам инвалидов, хосписам. Таким образом, благотворительность – наиболее общее понятие. И меценатство, и спонсорство могут рассматриваться как формы благотворительности.

Стремление к оказанию помощи тем, кто в ней нуждается, бесспорно, было присуще людям на протяжении многих веков. Основными формами меценатства и благотворительности на территории белорусских земель в период их вхождения в состав ВКЛ и РП являются примеры издания первых книг и создания первых типографий (деятельность Ф. Скорины, Н-Х. Радзивилла Чёрного, Н-К. Радзивилла Сиротки) [4], возведение храмов и монастырей (Н-К. Радзивилл Сиротка) [5], открытие учебных заведений (Н-К. Радзивилл Сиротка, М-К. Огинский) [6]. В период вхождения белорусских земель в состав Российской Империи были заложены основы новых форм благотворительности и меценатства, таких как коллекционирование археологических находок, создание нумизматических коллекций и библиотек с редкими рукописями, собирание древнего оружия, гобеленов, гравюр, картин, примером чего является благотворительная деятельность Эмерика-Николая и Кароля Яна Александра фон Гуттен Чапских [7].

Деятельность упомянутых ранее священнослужителей – Григория Шимановского в Дукоре и Петра Трусовского при Станьковской церкви, несмотря на отсутствие у них громких дворянских титулов, также дала свои плоды – привила заинтересованность в получении образования у крестьян, в результате чего сельское общество обновлялось людьми с новыми моральными устоями, что способствовало повышению духовности общества, формированию возвышенных моральных принципов у простых людей [8].

Таким образом, в ходе работы удалось определить основные тенденции развития благотворительности и меценатства на территории Беларуси, истоки которых берут начало еще в XVI веке. Благотворителями и меценатами при этом являлись не только магнаты, но и представители духовенства и других социальных групп [9]. История сохранила имена многих знаменитых личностей, которых мы и сегодня вспоминаем с благодарностью. Особая заслуга меценатов и благотворителей в том, что их стараниями были не только сохранены материальные ценности, но и положено начало образовательного процесса населения, что в свою очередь способствовало дальнейшему развитию благотворительности, как одной из высших форм проявления человечности.

К сожалению, память о меценатах прошлого мы не всегда бережем достойно. В этой связи необходимо больше популяризировать их вклад в историю страны, чаще говорить об их духовном подвиге и выборе, чтобы их результативные действия служили конкретным примером для подражания нашим современникам и не теряли своей актуальности в наши дни. «Чтобы процветало искусство, нужны не только художники, но и меценаты» – эти слова принадлежат знаменитому русскому театральному режиссеру прошлого века Константину Сергеевичу Станиславскому [10]. "Богатство обязывает", - высказывался более века назад банкир и предприниматель Павел Рябушинский. Думается, что эти идеи должны стать основой современного мировоззрения, которого так остро не хватает обществу массового потребления, страдающего от потери смысла своего существования.

В заключении хочется отметить, что наша страна богата добрыми, щедрыми и трудолюбивыми людьми. Многие благотворительные фонды («Шанс», «Юнихелп», «Белый аист»), образовавшиеся в начале этого столетия в Беларуси, оказывают большую помощь детям, нуждающимся взрослым, осуществляют поддержку реабилитационных программ именно за счет пожертвований широких слоев населения. При понимании этого факта возникает истинное чувство гордости за тех людей и организаций, которые продолжают славные традиции, заложенные много веков назад нашими предками, показывая всем нам, как можно стать настоящим человеком, осознающим и воплощающим в реальной жизни смысл таких главных гуманистических ценностей как истина, добро, красота, взаимная поддержка и радость от помощи ближнему.

**Список использованных источников:**

1. Михаленя, В.Р. Прикосновение к истокам Книга III / В.Р. Михаленя - Минск: УП «Энциклопедикс», 2017 – 124 с.
2. О предоставлении и использовании безвозмездной помощи: Указ Президента Республики Беларусь от 1 июля 2005г. №300: с изм. и доп.: текст по состоянию на 14 апр. 2020 г. – Минск: Дикта, 2005. – 11 с.
3. Буздуган, Я.М. Еволюція світового волонтерського руху / Я.М. Буздуган – Київ: Віче, 2013 - 10 С.
4. Малагевич, Е.С. Минск. Книга жизни / Е.С. Малагевич – Минск: Минская фабрика цветной печати, 2007 – 34 с.
5. Михаленя, В.Р. Храмы Дукоры / В.Р. Михаленя – Минск: А.Н. Вараксин 2014 – 65 с.
6. Летопись Дукорской Петро-Павловской церкви: 1867-1917 / Белорусский государственный архив-музей литературы и искусства - фонд 66, опись 1, дело 1156
7. Шибeko, З.В. Мінск у канцы XIX - пачатку XX стагоддзя / З.В. Шибeko – Мінск: Тэхналогія, 2009 – 105 с.
8. Кулажанка, У.Г. Летапіс Дукорскай Петра-Паўлаўскай царквы (1867-1917) / У.Г. Кулажанка – Мінск: Выдавецкі дом «Звязда», 2013 – 55 с.
9. Минские епархиальные ведомости: 1875 год № 3 / Национальный Исторический Архив Беларуси - с. 81-84
10. Станиславский, К.С. Моя жизнь в искусстве / К.С.Станиславский – США: Little, Brown and Company – 34 с.

## К ВОПРОСУ ОБ ОПРЕДЕЛЕНИИ КРИТЕРИЕВ НАУЧНОГО ЗНАНИЯ

Бурханов К.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Шкундич А.О. – преподаватель

Автор рассматривает вопрос определения научного знания, а также проблему разграничения науки и лженауки. Для этого автор раскрывает понятия терминов «наука» и «квазинаука». Предполагается, что особое внимание следует уделить практической значимости данных явлений, для чего необходим анализ потенциального вреда от лженаучных идей и гипотез.

Проблема демаркации (разграничения) науки и лженауки обсуждалась в философии на протяжении долгого времени, однако до сих пор остается актуальной. Демаркация – это принцип, позволяющий отделить научные теории от лженаучных. Пример одного из критериев научного знания – его универсализм. Так, оперируя универсальным научным знанием, любой математик при должном уровне подготовки сможет доказать теорему Пифагора, а химик – повторить опыт из учебника. Что нельзя сказать об астрологии и астрологах. Лженаука, как правило, имитирует научную деятельность и мимикрирует под настоящую науку.

Наука как социальный институт окончательно оформилась в XVII в. Научная революция и возросшие возможности применения научного знания поставили вопрос о разграничении науки и религиозно-мифологического знания. В первой половине XIX в. сформировалась теория позитивизма, в котором единственным источником научного знания провозглашался эмпирический опыт. Основателем позитивизма являлся французский социолог Огюст Конт. В первой половине XX в. сформировалось течение логического позитивизма или неопозитивизма, представители которого утверждали необходимость разработки научного языка и его логико-семантического анализа, чтобы устранить из него псевдонаучные метафизические высказывания. Именно для этой цели в Венском кружке (сообществе ученых, собиравшихся в Вене в 20-30-х гг. XX в.) был выдвинут принцип верификации высказываний, предназначенный для выполнения роли критерия демаркации – критерия, посредством которого разделялись псевдонаучные и научные высказывания. Согласно канонам этого принципа, высказывание может считаться научным, если его можно свести к так называемым атомарным высказываниям, которые фиксируют непосредственные опытные данные. Однако данный критерий было сложно применить к естественнонаучному знанию, поэтому поиск критериев демаркации продолжился.

Наиболее весомый вклад в решение проблемы демаркации внес австрийский ученый, основоположник философской концепции критического рационализма Карл Поппер. В основу его концепции заложен принцип фальсифицируемости той или иной гипотезы. Суть этого принципа заключается в том, что ученому всегда необходимо понимать, что должно произойти, чтобы высказанная им идея была неверна, иначе говоря, оказалась ложной. То есть предвидеть, какой результат эксперимента приведет его к отказу от собственной гипотезы.

Американский физик Томас Кун в своей работе «Структура научных революций» подверг критике релевантность верификации и фальсификации в качестве критериев научного знания. Кун противопоставлял этим критериям концепцию парадигмы. Он считал, что на развитие науки влияют не фальсификация и верификация, а существующая в конкретный исторический период парадигма.

Таким образом, можно выделить несколько определений науки:

- Наука – это специфическая деятельность по производству и получению новых знаний;
- Наука – это специализированная эмпирическая и теоретическая деятельность, направленная на получение истинного знания о мире;
- Наука – это деятельность, регулируемая идеалами и нормами получения, объяснения и построения научного знания.

Характерные черты научного знания: раскрытие сущностных, глубинных характеристик действительности; объяснение действительности; использование специальных терминов и раскрытие законов развития окружающего мира; предвидение событий, состояний предметов, которые произойдут в будущем; упорядоченность и использование определенных методов, например, метода эксперимента или наблюдения; наличие внутренней структуры.

Научное познание как процесс включает в себя два основных уровня: эмпирический и теоретический. На эмпирическом уровне осуществляется описание предметов, явлений и событий, на теоретическом уровне знания обобщаются, схематизируются.

Кроме разграничения научного и ненаучного знания, проблемой является различение ряда понятий, используемых для указания на ненаучность знания. Чаще всего термины «ненаука», «паранаука», «квазинаука», «антинаука», «псевдонаука», «лженаука» и им подобные употребляются взаимозаменяемо, имеются попытки последовательного разграничения этих понятий.

Лженаука – это теоретическая конструкция, содержание которой, как удается установить в ходе независимой научной экспертизы, не соответствует ни нормам научного знания, ни какой-либо

области действительности, а ее предмет либо не существует в принципе, либо существенно сфальсифицирован.

Б.И. Пружинин, доктор философских наук, главный редактор журнала «Вопросы философии», высказался по данному вопросу следующим образом: «Деятельность, претендующая на статус научной, может быть квалифицирована как псевдонаучная лишь тогда, когда появляются серьезные основания полагать, что действительные цели этой деятельности не совпадают с целями науки, что она вообще лежит вне задач объективного познания и лишь имитирует их решение» [2].

Можно классифицировать лженауки по критерию цели. Это деление является условным, поскольку одно и то же учение может быть направлено на разные цели, в зависимости от приверженцев этих лженаук, и уровня развития этих учений.

1. Лженауки первого типа не стремятся к выгоде. К этому типу можно отнести бесперспективные концепции, религиозные учения, а также многочисленные концепции разных самоучек, псевдоученых, стремящихся к прославлению, либо идеи людей, страдающие психическими отклонениями (шизофрения, маниакальные синдромы и т.п.), которые оказываются либо близкими к бреду, либо пустыми рассуждениями.

2. Лженауки второго типа стремятся к получению прибыли от частных инвесторов или торговли. Получение прибыли от людей осуществляется путем предоставления мнимых услуг и/или товаров, не обладающих заявленными действиями. Лженауки этого типа направлены на создание технологий промышленного применения или учений, которые могут заинтересовать распорядителей фондов и частных инвесторов. Также к этому типу относятся научные фальсификации, которые появляются в рамках официальной науки, в виде различных учений, чаще в актуальных направлениях. Они рассчитаны на получение выгоды от грантов или иного финансирования «научной деятельности».

3. Организованные (наиболее опасные и влиятельные) лженауки. Они направлены на получение особо крупной прибыли от государственных фондов, крупных, частных инвесторов и иностранного финансирования. Лженауки этого типа стремятся к влиянию на власть и науку сверху, через политические или академические структуры. Лженауки этого типа организованы в институты и академии, обладают устойчивым финансированием и влиянием. В этой форме лженаука переходит в квазинауку.

Отвечая на вопрос об определении научного знания, целесообразно выделить несколько устойчивых критериев, которые позволяют определить границы научного познания и отличить его от лженаучного. В качестве таких критериев можно выделить следующие: объективность, предметность, внутренняя непротиворечивость, доказательность. Таким образом, лженаука – это частный случай проявления в обществе такого явления, как ненаука, которая включает в себя как идеи, основанные на донаучном знании, псевдонауке, так и концепции, опирающиеся на мифологическое и религиозное мировоззрение (вера в колдунов, магию, экстрасенсов, астральные тела, ауры и т.д.). Псевдонаучное знание может причинить не только материальный вред, но и вред здоровью (гомеопатия и волновая генетика).

#### **Список использованных источников:**

1. Александров Е.Б. Феномен лженауки в современном обществе и меры по противодействию лженаучным проявлениям // В защиту науки. Бюллетень. 2014. № 13–14. С. 9.
2. Хазен А.М. О лженауке, ее последствиях и об ошибках в науке // Наука и жизнь. 2002. № 10. С. 103.
3. Эйдельман Е.Д. Ученые и псевдоученые: критерии демаркации // Здравый смысл. 2004. № 4 (33). С. 15.

## **ВЛИЯНИЕ ФЕНОМЕНА ТУЛЬПЫ НА СОВРЕМЕННУЮ ФИЛОСОФИЮ**

*Вайзин Н.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Мащитко С.М. – канд. философских наук*

В работе исследуются основные тенденции развития, происхождение, проблемы феномена тulpы. Автор выдвигает предположения о потенциале для исследования феномена и соответствующего сообщества с точки зрения философии, определяет основные вопросы, оказывающие влияние на сообщество и философию как науку в целом и излагает основные теоретические сведения, необходимые для решения вышеуказанных задач.

Феномен тulpы – социальное, психологическое и нейрофизиологическое явление, набирающее популярность в современном мире. Исследование данного феномена способно дать ответы на некоторые вопросы современной философии и психологии.

Исторически практики и случаи создания тulp начинают своё существование из культуры

монахов древнего Тибета, где тень воспринималась как дух, вызванный в мир людей и временный помощник. Современное же представление берёт своё начало из форумов и социальных сетей и годом первого упоминания считается 2005.

Тень - автономная виртуальная личность, способная проявляться как устойчивая самовнедрённая комплексная галлюцинация.

Существует постоянно растущее количество исследований, опросов и др. статистических данных, подтверждающих, что тень не является опасным для общества или индивидуума феноменом и не может рассматриваться, как психическое расстройство.

Согласно данным ценза сообщества, а также многочисленным записям отдельных практиков теньводства, в 95.6 % случаев появление тени улучшило общее качество жизни людей и/или принесло позитивное влияние на течение каких-либо психических заболеваний для хостов, уже подверженных им на момент создания (в 78% случаев).

В современном сообществе широко распространены метафизический («жреческий») и рационалистический («прагматический») взгляды на тень как явление.



Рисунок 1 – примерное представление спектра различных точек зрения на явление. Линия столкновения выражает невозможность пересечения данных взглядов.

Проблемы тени в философии:

#### 1. Проблема сознания

Ведущийся уже долгое время спор о справедливости учений монизма и дуализма может быть разрешен, если будет доказан или опровергнут феномен тени. Уровень жизнедеятельности тени изменяется с количеством внимания, взаимодействия с хостом и телом системы, которое она получает. Сознание и личность тени теряют целостность из-за долгого времени, проведенного в одиночестве, вплоть до полного исчезновения. Подобное ярко коррелирует с современными представлениями о нейрофизиологии человеческого мозга.

С точки зрения системы очевидно, что «человек» не привязан к телу. Тело для системы (а, следовательно, для всех людей) – это лишь носитель сознания, некий механизм для поддержания существования мозга – несомненно, истинного носителя «сущности» человека. Сознание неделимо на части, это единое целое для системы, то, с чем себя ассоциирует как тень(ы), так и хост, в то время как тело, бессознательное, примитивные реакции мозга (т.к. рефлексy), интуиция представляются совершенно не связанными с сознанием (другими словами, личностью) структурами и представляются чем-то отделённым от истинного «Я» объектов системы.

- В таком случае, можно ли сказать, что человек не тождественен своему организму?
- Или всё же, человек – это его мозг, как часть организма?
- Что есть «сознание»?
- Каким образом получается, что сознание может быть создано искусственно?
- Какие факторы, силы, феномены участвуют в процессе создания тени?

#### 2. Проблема искусственных форм жизни

Феномен тени можно рассматривать как первую успешную в истории человечества попытку создания искусственных форм жизни, обладающих самосознанием, способных принимать решения и иметь собственные мнения, паттерны поведения, взгляды, и обладающую сравнимым с человеческим интеллектом. Философская проблема гуманного отношения и правового и социального статуса искусственных форм жизни неоднократно попадала в поле зрения как научной философии, так и культуры, социума. Развитие сообщества носителей тени («теньводства», «хостов») выдвигает на более высокий план необходимость формирования конкретных стандартов, практик, правил и моральных и социальных норм в отношении как взаимодействия между тенью и хостом, так и внешне-социальных взаимодействий системы.

#### 3. Проблема сосуществования.

Согласно пониманию К.Г. Юнг[1], процесс форсинга (создания тени) имеет значительную корреляцию с процессом индивидуации личности. Различные (внешние, биологические, сознательные и бессознательные) факторы влияют на процесс формирования и протекания жизнедеятельности тени. Личность тени приобретает спектр различных черт, которые закрепляют её автономность и независимость. Впоследствии, в системе наступает период этических,

философских вопросов, таких как права тульпы и ответственность хоста за их выполнение. Собственные, не коррелирующие (а часто противоположные) желания и интересы тульпы и хоста заставляют каждую систему (совокупность всех объектов одной физической единицы) искать способы решения. Известны как случаи крайнего повышения/понижения статуса тульпы (заменить хоста в управлении материальным телом, или наоборот, являться инструментом для удовлетворения личных желаний хоста), так и случаи баланса, различные по положению на спектре тульпа – хост, где каждый имеет право свободно манипулировать общим телом и взаимодействовать с внешним миром, в течение ограниченных промежутков времени. Таким образом, формируются следующие трудноразрешимые вопросы:

- Является ли личность тульпы «настоящей»?
- Откуда на самом деле берёт своё начало эта личность?
- Имеет ли тульпа право на жизнь, личное время, интересы, общение – грубо говоря, имеет ли она право «на тело», настолько ценный и экзотический ресурс для этого существа?
- Каковы социальные права тульпы?
- В какой мере применимы моральные нормы к личности тульпы?
- Можно ли считать прекращение доступа тульпы к хосту и внешнему миру её смертью?

Список использованных источников оформляется в соответствии с требованиями ВАК РБ по категориям. Требования размещены в свободном доступе на [vak.gov.by](http://vak.gov.by). Заголовок выделяется жирным шрифтом (см. ниже). Перед списком остается пробельная строка.

**Список использованных источников:**

1. Юнг К. Г. Тэвистокские лекции. Аналитическая психология: её теория и практика / пер. с англ. В. И. Менжулина. — М: АСТ, 2009. — 1-252 с.
2. Varieties of Tulpa Experiences: The Hypnotic Nature of Human Sociality, Personhood, and Interphenomenality : заметки о киберэтнографических исследованиях / Veissiere, Samuel. Oxford University Press, 2015 – 5-16 с.
3. Arthur Juliani, On Tulpas : An analysis of imagined others : заметки об исследованиях / Arthur Juliani, University of Oregon – 1-20 с.
4. Tulpas and Mental Health: A Study of Non-Traumagenic Plural Experiences / Jacob J. Isler, Department of Communication, University of Texas at Austin, Austin, Texas, United States, 2017 – 36-43 с.
5. "What is a Mental/Psychiatric Disorder? From DSM-IV to DSM-V". / Stein, Dan J; Phillips, K.A; Bolton, D; Fulford, K.W.M; Sadler, J.Z; Kendler, K.S. Psychological Medicine, 2010 - 40 с. (Вольный перевод фрагмента)
6. [Tulpa.info](http://tulpa.info): интернет-ресурс (посещался в 2019 г.)
7. Kiahdaj's Absolute Guide to Tulpas : электронная литература / Автор неизвестен. (посещался в дек. 2019) - 22 с. [https://docs.google.com/document/d/17iiFxZ9PYXXxSqWHdU2YoPsGfJr90w8aIG\\_evqc1wiY/edit](https://docs.google.com/document/d/17iiFxZ9PYXXxSqWHdU2YoPsGfJr90w8aIG_evqc1wiY/edit)

## **СУБЛИМАЦИЯ И ЦИВИЛИЗАЦИЯ В ПСИХОАНАЛИЗЕ**

*Верига М. Е.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Бархатков А.И. - кандидат философских наук*

Сублимация – термин, введенный Зигмундом Фрейдом для обозначения защитного механизма психики, посредством которого энергия либидо перенаправляется в общественно и лично безопасное русло. Согласно Фрейду, именно механизм сублимации лежит в основе любой творческой деятельности и поэтому является основой всей человеческой цивилизации.

Каждый день нам приходится испытывать различные желания и принимать решения, которых требуют от нас обстоятельства и наши побуждения. Побуждения эти, однако, бывают и положительными, и отрицательными, и каждый человек должен уметь справляться с любыми из них. Одним из способов совладать с ними является процесс сублимации.

Сублимация в переводе с латинского «sublimare» буквально означает «замещать», «подменять». Впервые этот термин употребил Зигмунд Фрейд. Он был впечатлен книгой Генриха Гейне о мальчике, который в детстве издевался над собаками, а позже стал уважаемым хирургом. Фрейд назвал это сублимацией и описал как один из защитных механизмов психики, который помогает избавиться организму от чрезмерного напряжения, которое сам человек не всегда может осознать и снять. Также он предположил, что именно та энергия, которая привела ребенка к садизму, в итоге была сублимирована на действия, приносящие окружающим пользу.

В работах Фрейда речь шла в основном о либидо – неразряженном сексуальном напряжении, которое приводит индивида к творческой деятельности. Проще говоря – испытывая, но не имея возможности реализовать сильное половое влечение, человек начинает рисовать, петь, танцевать, готовить, шить и т.д. Таким образом, сублимировать означает подменять способ реализации своей потребности. Нередко это происходит, когда человек (возможно, неосознанно) хочет чего-то

стыдного, осуждаемого обществом, идущего наперекор морали, религии или уголовному кодексу.

Согласно психоаналитической теории Фрейда, личность состоит из трех компонентов. Первый из них, Ид (Оно), служит источником либидо – энергии, стимулирующей поведение. Ид состоит из всех примитивных и простых желаний и побуждений, направленных на получение удовольствия и в большинстве своем неприемлемых для общества. Вторая составляющая личности, Эго (Я), формируется позже, в детстве, для управления Ид. Для этого Эго переводит бессознательные желания в форму, соответствующую требованиям реальности и заставляет нас справляться с ними другими способами. Третий компонент личности, Супер-Эго (Сверх-Я), заключает в себе все моральные нормы, правила и ценности, усвоенные через родителей и собственный опыт жизни в обществе, следит за их соблюдением и регулирует поведение человека.

В силу такой сложной структуры, Эго приходится выступать посредником между требованиями реальности, побуждениями Ид и соблюдением морали Супер-Эго. Но поскольку удовлетворить все запросы редко представляется возможным, Эго страдает от чувства тревожности. И в этом случае сублимация – настоящее спасение от нее, направляющее любые внешние импульсы в приемлемое или даже поощряемое обществом поведение. Как итог, аффект разряжается без нарушения личных и общественных правил.

Типичные примеры сублимации:

- человек с садистическими наклонностями и влечением причинять боль другим становится хирургом;
- человек с высокой сексуальной энергетикой выражает себя через искусство, например, художество или танцы с эротическими элементами;
- человек, внутри которого кипят отрицательные эмоции, нередко начинает убирать в своем жилище или готовить пищу, усердно орудуя ножом;
- человек, который занимается спортом и участвует в соревнованиях ради получения радости и гордости от чувства победы и доминирования.

Фрейд считал способность к сублимации признаком зрелости, так как именно она позволяет людям вести себя благоразумно. Также им было высказано предположение, что сама наша цивилизация, возможно, возникла благодаря сублимации, ведь именно этот процесс заставляет людей заниматься творческой и созидательной деятельностью.

Любимым примером Фрейда был Леонардо да Винчи, великий живописец, учёный и инженер. Его вклад в развитие культуры и науки трудно переоценить. Он доводил до конца любое дело и, если начинал работать в какой-то сфере, то делал это до тех пор, пока не достигал совершенства. Зигмунд Фрейд объяснял эту особенность да Винчи полной сублимацией его сексуальной энергии. Свою необычайную работоспособность Фрейд также объяснял как результат полной и сознательной сублимации половой энергии уже в сорокалетнем возрасте.

Таким образом, сублимация – здоровый и зрелый способ борьбы с нежелательными или недопустимыми побуждениями, потому что, не давая нам навредить своими действиями себе или другим людям, она помогает распределить энергию на более полезные дела. Можно сказать, что, независимо от выбранного способа сублимации, можно быть уверенным в продуктивности и положительности ее результатов.

## ТРАНСГУМАНИЗМ. НА ПУТИ К ПОСТЧЕЛОВЕКУ

Гурец Н.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Пушкина Т.А. – к.ф.н., доцент

Трансгуманизм стремительно набирает популярность в последнее время во всем мире уже не как футуристический проект, а как реальная философско-политическая сила. В работе рассмотрены основные аспекты трансгуманизма, его цели и задачи, а также дан развернутый ответ на вопрос – что же это такое, *трансгуманизм?*

Трансгуманизм – философская концепция, а также международное движение, поддерживающие использование достижений науки и технологии для улучшения умственных и физических возможностей человека, с целью устранения тех аспектов человеческого существования, которые его последователи считают нежелательными — страданий, болезней, старения и смерти. Иными словами, трансгуманизм предлагает человечеству обрести свободу от биологических ограничений вида. В этом трансгуманизму способствуют новейшие разработки в области генной инженерии, кибернетики и нейрохирургии.

Многие трансгуманисты (в частности, Рэймонд Курцвел) считают, что непрерывно ускоряющееся техническое развитие уже к 2050 году приведет к явлению технологической

сингулярности[1] – событию, после которого ключевые научные открытия будут совершаться практически одновременно. Человек уже не сможет сохранять свою власть над стремительно меняющимся миром. Здесь на помощь приходит трансгуманизм, который позволит создать постчеловека, способности которого будут принципиально отличаться от способностей современных людей.

Сторонники трансгуманизма открыто поддерживают любые достижения науки и техники и доводят идею прогресса до ее логического конца. Символ трансгуманизма – латинская буква H от «human» – то есть человеческое, и значок плюс – “H+”. H+ – это постчеловечество, свободное от болезней, несовершенств, и которое в конце концов достигнет физического бессмертия.

Однако трансгуманизму все еще необходимо разрешить некоторые серьезные задачи:

- 1) Изучать и внедрять научные и технические достижения;
- 2) Устранять любые опасности и проблемы (в том числе и нравственные), мешающие внедрению инноваций и улучшению человека.

Постчеловек является одним из немногих путей выживания человечества в долгосрочной перспективе. И действительно, благодаря новым технологиям будет достигнуто изобилие в территориях и ресурсах: войны и насилие потеряют свой смысл. Замена слабых тел из плоти и крови совершенными механическими аналогами позволит человеку совершить эволюционный прорыв. Более не будут страшны болезни, старость и смерть. Однако существуют серьезные препятствия для такого возвышения человека экзистенциального характера [2]. Но как физическое бессмертие влияет на человека? Готово ли общество к эпохе киборгов, мутантов и химер?

Одни из немногих, кто остается по другую сторону идеологического фронта от трансгуманизма – это фундаментальные традиционалисты. Они же и не воспринимают технологический прогресс как становление к лучшему. В их понимании современное общество – это не итог прогресса, а результат упадка.

В священном писании говорится, что Дьявол не способен сотворить человека, он может лишь спародировать его – создать его симулякр. Разве трансгуманизм – это не его затея?

**Список использованных источников:**

1. KurzweilAI – (Forbes | Human 2.0: is coming faster than you think).
2. Хайдеггер: предел нисхождения Dasein'a (Лекция 8). МФК. 2013. Социологический факультет МГУ. Лекция от 30/10/ Dasein (mayrefuse). Экзистенциальное время. Начало и конец европейского Логоса. Gestell. Выбор (Entscheidung).

## **ФИЛОСОФИЯ ВЫБОРА ИНДИВИДА**

*Голубев Владислав Алексеевич*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Шепетюк В.В. – кандидат философских наук*

В данной работе исследуется значение выбора в жизни индивида, который рассматривается в качестве субъекта выбора. Рассматривается типология выбора и область использования как ситуации выбора в целом, так и отдельных типов решений субъекта выбора. С помощью приведенных в тексте пояснений представлен возможный пример определения выбора как феномена жизни человека. Рассматривается эксперимент по выбору субъекта из двух совершенно одинаковых альтернатив, анализируются результаты и подводится итог моделирования ситуации выбора.

Проблема выбора актуальна на протяжении всего времени существования человека. Ведь сам жизненный цикл его существования можно представить как адаптированную систему, состоящую из огромного множества ситуаций выбора. Выбор можно рассматривать как условно осознанное субъектом действие, направленное на устранение неопределенности в его существовании.

Понятие выбора может быть классифицировано следующим образом.

1) Простой выбор. Это обычное сравнение объектов выбора по альтернативному критерию. Если разница между показателем характеристики очень мала или наоборот очень велика, то выбор часто может оказаться очевидным.

2) Смысловой выбор. Когда критерием для сравнения альтернатив не даны изначально и человеку необходимо их конструировать (например, выбор супруга, профессии и т.д.). Субъекту необходимо найти общие основания для сопоставления качественно разных альтернатив и сформулировать критерии их оценки. Перед субъектом в этом случае стоит задача по установлению смысла, то есть задача определить для себя смысл каждой из альтернатив.

3) Личностный (экзистенциальный) выбор. Это жизненно важный выбор в критических жизненных ситуациях, когда субъекту не даны ни критерии для сравнения альтернатив, ни сами альтернативы.



Здесь встает необходимость самостоятельного конструирования альтернатив и на основании предположения о том, что будет при выборе одной из альтернатив в будущем, возникает необходимость делать свой выбор.

Наряду с указанной классификацией, выбор можно рассматривать как осознанный, когда мы полностью уверены в своём владении информацией об объектах выбора и отдаём себе отчет о возможных последствиях, а также как неосознанный, когда мы не владеем достаточной информацией и не можем предвидеть последствия.

При любых возможных классификациях и подходах следует помнить, что выбор является преимущественно активным действием субъекта, а не пассивной реакцией. Основанием выбора не может являться лишь побуждение как таковое. Там, где дело решается исключительно силой внешнего воздействия и принуждением, нет смысла говорить о выборе, согласно утверждениям некоторых психологов. Иными словами, речь идет о выборе в возникшей ситуации т.н. отсутствия выбора. При этом ряд специалистов все же утверждает, что даже если на субъекта выбора чрезмерно влияют внешние факторы, выбор все же останется, однако последствия этого выбора изменятся, т.к. субъект всегда имеет право пойти против влияющей силы, несмотря на то, что последствия, возможно, будут не такими как предполагалось изначально. Примечательно, в этой связи, что в данном случае в качестве субъекта выбора уже может рассматриваться не только сам участвующий непосредственно индивид, но и субъект (-ты), сознательно формирующие для участника предлагаемые взаимоисключающие альтернативы. Еще одной совершенно иной и крайне интересной формой выбора является ситуация действия при наличии двух совершенно идентичных объектов выбора, по типу т.н. «Буриданова осла», когда необходимость решения может показаться абсолютно бессмысленной. В этой связи особый интерес вызывает исследование деятельности субъекта выбора с точки зрения того, чем он руководствуется для решения той или иной задачи, связанной с необходимостью выбора.

Как уже было сказано, с неизбежностью выбора связана очень большая часть жизни человека. С одной стороны, когда мы выбираем, мы постепенно начинаем овладевать ситуацией, изучаем ее, рассматриваем с разных сторон и, таким образом, как бы мы адаптируемся под окружающий нас мир и одновременно адаптируем по возможности отдельные стороны мира под свои нужды. С другой стороны, если представить себе, что в жизни человека исчез выбор, то людям не пришлось бы заниматься мозговой и интеллектуальной деятельностью по поводу решения задач, связанных с выбором, а ведь такие задачи иногда невероятно сложные и мы просто не в состоянии решить многие из них, в связи с чем возникает вопрос – зачем они вообще нужны, если решения нет, есть ли в них какой-то иной смысл?

Выдающийся русский философ и культуролог XX в. Михаил Михайлович Бахтин своими исследованиями показал, что бытие человека или другого разумного существа не представляется возможным без поступков, связанных с ситуацией выбора. Другими словами, сутью любой сознательной деятельности и есть осуществление выбора. Важным при этом является не только и не столько сам факт выбора, как основа любой сознательной деятельности, но также и глубоко ответственное отношение человека к результатам своего выбора. Именно такая последовательность действий создает в итоге человека в полном смысле этого слова, для которого М. Бахтин вводит особую поведенческую формулу «не алиби» в бытии [1]. Последнее, по его мнению, объясняется тем неустранимым фактом, что человек всегда неизбежно отвечает за результаты своей деятельности и своего выбора, а, следовательно, не должен и фактически не может не поступать правильно, как того требует его разум и совесть. Несмотря на имеющиеся вопросы о критериях правильности выбора нельзя не отметить глубокий смысл и гуманистический потенциал взглядов русского мыслителя относительно природы выбора человека, как основном механизме его самосовершенствования.

Рассмотрим эксперимент по выбору из двух совершенно одинаковых альтернатив. Ранее мы выяснили, что субъект выбора из двух совершенно одинаковых вариантов начинает выбирать исходя не из фактических объектов выбора, а из перечня влияний окружающей среды на него либо на объекты выбора. Примером этого является простейшая ситуация, когда на какой-то объект в большей степени падает свет, вследствие чего появляется большая вероятность, что субъект выбора склонится к данному объекту. Итогом такого эксперимента будет являться утверждение о том, что мы выбираем, в сущности, исходя не из собственных критериев, а из совокупности свойств предмета выбора, среди которых можно назвать т.н. адаптивность и эластичность, которые часто используются для описания процессов в экономике и психологии [2].

Таким образом, мы можем заметить, что в ситуации, когда мы совершаем выбор, как нам кажется, по собственному убеждению, мы фактически лишь следуем логике вещей, которая может быть изначально задана. В некотором смысле при таком понимании наш выбор и участие в процессе выбора оказываются минимальными, а сам выбор предстает лишь иллюзией.

В науке существует еще одно одноименное понятие – адаптивный выбор, т.е. выбор, сделанный в пользу одобряемых действий или соответствующий общепринятым нормам. Если обращаться к

большинству случаев такого выбора, то можно отнести его к пассивным реакциям индивида, выражающейся в следовании тому, что одобряется обществом и окружающими. Вместе с тем для многих представителей, например, молодежи, свойственны проявления т.н. контрадаптивного выбора, когда действие осуществляется, напротив, вопреки общественной оценке, привычным понятиям плохого и хорошего, достойного и порицаемого. В таком случае контрадаптивный выбор можно рассматривать как попытку молодого человека заявить о себе и о своей уникальности, как попытку конструирования своего уникального «Я» и элемент процесса самоидентификации, что требует проверки в ситуациях различного, порой весьма противоречивого выбора. В тоже время такой поиск себя и связанный с ним выбор может рождать и новое понимание стандартов социального поведения, ведь бездумное следование наиболее типичным образцам поведения часто лишь разрушает здоровую ткань социальной реальности, когда индивиды массового выбирают то, за что их похвалят, однако сами в важность такого выбора не верят. В подобной ситуации, указанная еще М. Бахтиным важная составляющая выбора – ответственность, разумеется, отсутствует, что может привести к определенной деградации социальной системы.

Современный человек обладает достаточными интеллектуальными способностями, чтобы сделать осознанный выбор, однако в постоянно усложняющемся мире так происходит далеко не всегда. Правильный выбор ассоциируется у индивида с чувством контроля над ситуацией и властью над обстоятельствами, в силу чего множество организаций и социальных институтов стараются предоставить индивидам множественный выбор, целью которого может быть простое создание иллюзии влияния и широты возможностей, что способно привлечь максимальное количество людей и заинтересовать их. Такая массовая заинтересованность выступает важнейшей основой большинства используемых маркетинговых и рекламных стратегий продвижения товаров и услуг, создания условий для управляемого и прогнозируемого выбора клиентов.

В заключении хотелось бы отметить, что выбор играет в жизни человека роль начальной точки любого действия, тем самым характеризует себя как жизненно важный процесс. Благодаря нашему исследованию вопросов выбора мы выяснили, что человек выбирает подсознательно анализируя совокупность свойств объектов связанных с адаптивностью и эластичностью совершаемого выбора, то есть выбирает, обращая большее внимание на значительное количество внешних факторов, которые он не в состоянии контролировать, но которые очень важны для него, а, значит, тема выбора есть и будет одна из самых актуальных тем в его жизни.

**Список использованных источников:**

1. Бахтин, м. М. К философии поступка // философия и социология науки и техники. Ежегодник 1984-1985. – м., 1986.
2. Карпович, а. И. Характеристики устойчивости экономической системы и их взаимосвязь // науч. Ст. По специальности «экономика и бизнес» // нгузу, 2018.
3. Шапарев, в. П. Психология рациональности в аспекте выбора как фактор процесса принятия решений // науч. Ст. // кну им. Ж. Баласагына, 2017.

## **ФИЛОСОФИЯ АЛЬБЕРА КАМЮ В РОМАНЕ ЧАКА ПАЛАНИКА «БОЙЦОВСКИЙ КЛУБ»**

*Зотова П.П.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Шелетюк В.В. – кандидат философских наук*

Цель исследования – раскрыть особенности философии Альбера Камю, нашедшие художественное отражение в романе Чака Паланика «Бойцовский клуб». В работе рассмотрены основные вопросы философии Альбера Камю, которые актуальны для общества двадцать первого века, раскрыта сущность таких понятий, как «антиконсюмеризм», «потребительство» и «общество спектакля» в совокупности их принципов и форм. В результате определен сценарий, при котором субъект борьбы с глобальным потребительством сам становится его активным участником.

Зачастую в произведениях определённого периода отражены проблемы, с которыми сталкивался социум непосредственно в момент возникновения идеи и реализации замысла этого произведения. Поэтому людям XXI века интересны тексты, написанные в конце девяностых годов двадцатого – начале нулевых годов двадцать первого века.

Особенно показательными, на наш взгляд, в этом плане выглядят романы Чака Паланика. Этот современный американский автор стремится заставить своих читателей испытывать самые разнообразные чувства: от удивления и восторга – до испуга, отвращения и даже отторжения. Его культовый роман – «Бойцовский клуб», вышедший в свет в 1996 г., является своеобразной критикой общества нашего времени – общества тотального потребления.

В романе отражены 12 наиболее актуальных, по мнению автора произведения, проблем современного мира. Среди них анархизм, гендерное неравенство, проблемы личностной

самоидентификации, капитализм, насилие, упадок массовой культуры и многие другие. Главный герой произведения – типичная отвергнутая обществом личность, страдающая девиациями, офисный клерк, старающийся выбраться из паутины потребительского существования и однообразной реальности.

Принято считать, что на каждое действие всегда может найтись противодействие. Так и у идеологии потребления, отвергаемой героями романа, появилась совокупность общественных, экономических, культурных и контркультурных движений, направленных на ее критику. При этом проблема, поднятая в произведении настолько глубока, что стало необходимо разграничить естественные потребности и навязанные обществом, корпорациями, СМИ и избавиться от искусственных, уродующих человеческую жизнь мотивов деятельности. По сути, первой страной, где возникло антипотребительское движение, стали США – родина рассматриваемого романа Ч. Паланика. Таким образом, возникшая идеология антиконсюмеризма – своеобразная рефлексия общества потребления над самим собой. Ч. Паланик лишь подметил сопутствующие ему явления современной общественной жизни – цинизм, информационные манипуляции и обман населения, долговое рабство, внушение страхов и комплексов несоответствия, взаимная неприязнь, тотальная конкуренция. Многие черты подобного рода разрушительного мировоззрения отражены в одной из самых узнаваемых цитат «Бойцовского клуба» – «Поколения за поколениями люди работают на ненавистных работах только для того, чтобы иметь возможность купить то, что им не нужно» [1].

Но что такое настоящий антиконсюмеризм? Самокритика, новый циничный самообман или очередная попытка сбросить с себя груз накопившихся мировоззренческих проблем и противоречий? Для анализа данного вопроса следует использовать еще одно близкое понятие «общества потребления» понятие «общество спектакля», введенное в оборот уже французским философом и писателем Ги Дебором [2], для отражения невозможности отличить искренность от обмана в условиях господствующей культуры. Тем самым можно зафиксировать совпадение взглядов Ч. Паланика с другими авторами относительно описания доминирующей на сегодняшний день негативной социальной реальности, необходимости ее коренного изменения.

Главный герой романа «Бойцовский клуб» полагает, что современное общество потребления подавляет природное мужское начало, превращая мужчин в нечто пассивное. И здесь мы видим соответствие идеям философии известного французского философа и публициста А. Камю, когда «просто человек» сталкивается с обществом, которое принудительно «каталогизирует» каждого. При дальнейшем знакомстве с романом «Бойцовский клуб» мы находим и другие основные аспекты философии Альбера Камю, основной из которых – феномен бунта, восстания, главная цель которого – преобразование жизни, объединение личности и социума [3]. Такой бунт должен повторяться до тех пор, пока не будет установлен новый порядок, где противоречия предыдущей культуры будут устранены. В романе «Бойцовский клуб» подобный тезис отражен следующим образом: рассказчик, от имени которого ведется повествование в романе, бунтует против начальника, общества, Бога.

В свою очередь рабская зависимость от изобилия, описанная в романе – также отражение взглядов А. Камю, который считал современное общество – обществом рабов и господ. Метод решения этой проблемы, согласно А. Камю, выглядит анархо-фашистским: добровольно уничтожить свою собственность и насильно – чужую, попытка чего и осуществляется героями романа «Бойцовский клуб».

Рассматривая другую проблему, поднятую в романе – проблему взаимоотношения личности и социума, следует помнить, что «абсолютная власть закона не есть свобода, но не большей свободой является абсолютная неподвластность закону. Расширение возможностей не даёт свободы, однако отсутствие возможностей есть рабство. Но и анархия тоже рабство. Свобода есть только в том мире, где четко определено как возможное, так и невозможное» [3]. Однако «сегодняшний мир, по всей видимости, может быть только миром господ и рабов», ведь только рабское сознание, по мнению А. Камю, творит историю.

Итогом романа является утверждение мысли о том, что только культ насилия может победить культ потребления. В этом смысле роман «Бойцовский клуб» – это наглядный пример уже упомянутой идеи Ги Дебора о том, что «спектакль – это стадия, на которой товару уже удалось добиться полного захвата общественной жизни» [2]. Копировальный аппарат, с которым так часто сталкивается главный герой произведения в мире своей повседневности, – один из типичных символов консюмеризма, раскрывающий содержание культуры как результат тотального тиражирования и конвейерного производства всего того, что пользуется спросом. В этом смысле наш интерес к роману «Бойцовский клуб» заключается ещё и в том, что книга, призванная разоблачить общество потребления, сама становится объектом потребления, а протест против системы бездумного всеобщего повторения и воспроизведения всего наиболее популярного, сам же лишь укрепляет основы существующего порядка вещей. На наш взгляд, в этом видится яркое воплощение идеологии цинизма. Возможно именно поэтому многие персонажи романа Ч. Паланика, созвучно тезисам А. Камю, парадоксальным образом идеализируют фашизм как выход из состояния общества всеобщего потребления.

Таким образом, идеи романа Ч. Паланика «Бойцовский клуб» крайне актуальны на сегодняшний день и находят многочисленные пересечения не только с философскими взглядами А. Камю, но и многих других авторов, размышляющих над проблемами жизнедеятельности современного общества. Безусловно, должны быть найдены какие-то методы изменения доминирующих взглядов, ценностей, вкусов и предпочтений, приемлемые для своеобразного перевоспитания больших масс населения. Однако все ещё предстоит определить, какие именно инструменты способны наиболее позитивно воздействовать на поведение больших групп людей, не порождая при этом бездумное повторение, тиражирование популярных трендов и неизбежно возникающее вслед за этим потребление. А пока спектакль всеобщего «потребления» жестко контролирует человеческое общество и диктует ему свои законы поведения. Чтобы решить это сложное противоречие, необходимо понять, что усиленное распространение культуры демонстративного потребления всегда будет порождать демонстративное антипотребительство, как попытку стабилизировать ситуацию и найти равновесие в условиях болезненного переживания всеми нами – потенциальными потребителями – невозможности полного удовлетворения своих искусственно создаваемых и навязываемых потребностей.

**Список использованных источников:**

1. Паланик, Ч. Бойцовский клуб / Ч. Паланик; пер. К. Егоровой. – Москва: АСТ, 2018. – 256 с.
2. Дебор, Г. Общество спектакля / Г. Дебор; пер. С. Офертаса и М. Якубович. – Москва: Логос, 1999. – 224 с.
3. Камю, А. Бунтующий человек. Философия. Политика. Искусство / А. Камю; пер. И. Я. Волевич, Ю. М. Денисов, А. М. Руткевич, Ю. Н. Стефанов. — Москва: Политиздат, 1990. — 415 с.
4. Шамина, В.Б. Романы Чака Паланика «Бойцовский клуб», «Уцелевший», «Удушье» в контексте романтической традиции / В.Б. Шамина // Ученые записки Казанского университета. Сер. гуманитарные науки. – 2012. – № 2. – С. 189-200.
5. Горохов, П.А. Некоторые мысли о философии Альбера Камю / П.А. Горохов // Вестник Оренбургского государственного университета. Сер. философия, этика, религиоведение. – 2013. – № 7. – С. 4-9.

## **«МЕМ» КАК СМЫСЛООБРАЗУЮЩИЙ ИНСТРУМЕНТ ИНФОРМАЦИОННОЙ КОММУНИКАЦИИ**

*Косякова Е.В., Никитенко Е.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Лагунова Е.Н. – преподаватель*

Информационно-коммуникативные технологии являются определяющим фактором развития современного общества. Человек разумный приобретает все больше облик «человека информационного». В работе рассматривается понятие «мема» как одной из характеристик массового человека в информационном пространстве.

Исторически наука представляет социокультурный феномен, созданный человечеством в процессе своего развития. Основой науки является разум, который определяет и создает, во многом, человека и культуру как неприродную среду обитания и социального взаимодействия. Результатом такой «рационализации» является ситуация, когда наука с ее неограниченными возможностями и культом разума, столкнулась с теми проблемами, которые сама же и взрастила. Действительно, гонка вооружений, международный терроризм, всплеск региональных и межконфессиональных конфликтов, «цунами» миграции — вот далеко не полный перечень угроз и трагедий нашего века.

Наука, смысловой центр человеческой культуры, вывела человечество на грань катастрофы. Стало ясно, что она создала возможности быть используемой не только для человека, но и против него. Одним из феноменов, требующим рассмотрения, сегодня выступают информационно-коммуникативные технологии.

На современном этапе развития человечества усиливается противоборство в глобальном информационном пространстве, обусловленное стремлением некоторых стран использовать информационно-коммуникативные технологии для достижения своих геополитических целей. Путем манипулирования общественным сознанием, в том числе. Актуальность данной проблемы связана с поиском перспективных инструментов ведения информационного противоборства. Здесь представляется целесообразным обратить внимание на использование потенциала меметики как инструмента достижения поставленных целей.

Меметика находится на стыке вненаучного и научного знания и строится по аналогии с теорией эволюции Чарльза Дарвина. Теорию о кодировании культурной и поведенческой информации в мемах нельзя подтвердить исследованиями, опытами или экспериментами, но последователи меметики полагают, что вся культурная информация состоит из базовых единиц – мемов, точно так же как биологическая информация состоит из генов. Целью меметики является стремление теоретически обосновать культурную эволюцию человека на основе мемов как единиц культурной информации.

С учётом разночтений понятия мема, можно выделить его ключевые характеристики:

- в мемах проявляются глубинные образы мышления;
- мемы влияют на совершаемый человеком выбор;
- мемы могут иметь деструктивный и позитивный характер (например мистические мемы проявляются в качестве жертвоприношения и сказочной фантазии);
- мемы – это структуры мышления, т.е. они влияют на принятие решений, но не формируют убеждения и ценности личности;
- мемы изменчивы: они могут разгораться и тускнеть, приобретая законченный и понятный обществу смысл лишь в рамках конкретного исторического контекста.

Необходимо отметить, меметика может использоваться как для негативного, так и позитивного информационного воздействия, так и для формирования положительного образа продвигаемой идеи. С одной стороны меметика входит в сферу исследования и применения военных специалистов и политтехнологов (примером может служить «оранжевая революция» в Украине или «революция роз» в Грузии), с другой — мемы являются выражением стереотипов общества.

Современное информационное пространство управляется с помощью «меметических технологий», и с каждым новым витком развития человечество будет создавать всё более совершенные технологии влияния, основанные на манипулятивных методах. В этом смысле, меметика представляет собой перспективный инструмент информационно-психологической борьбы, а мем — один из интенсивных инструментов современной коммуникации, способных влиять на общественное мнение.

**Список использованных источников:**

1. Докинз, Р. Эгоистичный ген / Р. Докинз. — М. : Мир, 1993. — 318 с.
2. Лысенко, Е. Н. Интернет-мемы в коммуникации молодежи/ Е.Н. Лысенко // Вестник Санкт-Петербургского университета: Социология, 2017. — 10(4). — С.410-424.
3. Щурина, Ю. В. Интернет-мемы: проблема типологии/ Ю.В.Щурина // Вестник Череповецкого государственного университета: Филологические науки, 2014. — № 6 (59). — С. 85-89.

## **КОГНИТИВНЫЕ ИСКАЖЕНИЯ И ИХ ВЛИЯНИЕ НА ИНДИВИДУАЛЬНЫЙ ВЫБОР**

*Кукареко А.П.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Лагунова Е.Н. – преподаватель*

Каждый день люди принимают множество решений, полагаясь на определённые когнитивные схемы, что позволяет выносить правильные суждения. Но данные модели также приводят к когнитивным искажениям, ошибкам и нежелательным результатам.

Когнитивные искажения – систематические отклонения в восприятии, мышлении и поведении, обусловленные субъективными убеждениями (предубеждениями) и стереотипами, социальными, моральными и эмоциональными причинами, сбоями в обработке и анализе информации, а также физическими ограничениями и особенностями строения человеческого мозга. Автор понятия, Д. Канеман, различал два способа мышления [3]. Так, существует «быстрое» (автоматическое) мышление и «медленное» (осознанное) мышление. Медленное мышление предполагает формальное взвешивание вариантов, что является частью рационального принятия решений, что особенно полезно для установления точных связей между сложными концепциями. Недостатком такого мышления является то, что для медленного осознанного размышления требуется наличие свободного времени, конкретной информации и мотивации. Автоматическое мышление предполагает «фоновую обработку» подавляющего большинства решений, что значительно упрощает их принятие. Однако за скорость и эффективность быстрого мышления приходится платить: это может приводить к чрезмерному обобщению и пренебрежению некоторой потенциально важной информацией. Более 100 когнитивных искажений, в том числе прайминг-эффект, подчинение авторитету, искажение оценки по среднему, ошибка выжившего, эвристика доступности, ложная память, иллюзия конца истории, недооценка бездействия и другие были описаны в литературе, и исследования в области когнитивной и социальной психологии продолжают часто выявлять и определять новые предубеждения [1].

В ходе анализа тематических исследований было определено, что основными причинами различных инцидентов, аварий, столкновений или катастроф являются когнитивные искажения. Уменьшение предвзятости при принятии решений является одной из важных мер по предотвращению таких нежелательных событий. Отсутствие понимания человеческой иррациональности приводит к бессознательному повторению ошибок, что приводит к порочным кругам подобных критических

инцидентов. Например, ошибка конъюнкции и ошибка базового процента могут быть в некоторой степени подавлены с помощью изменения способа представления информации от вероятностного к частотному. Тодд и Гигеренцер [2] показали, что простые (быстрые и экономные) эвристики работают лучше по сравнению со сложными и алгоритмами в ситуациях, когда простота приводит к устойчивости.

Чтобы избежать когнитивных искажений или иррациональных и искаженных решений, возможно использование «контрольных листов» когнитивных искажений, когда при принятии важного решения производится дополнительная проверка информации. Если мы можем выявить когнитивные искажения и определить, где они могут быть присущи или вероятны в рамках конкретных систем, с помощью контрольного листа или накопленной статистики когнитивных искажений могут быть разработаны соответствующие меры безопасности или контрмеры, которые снизят вероятность ошибок при принятии решений.

В настоящее время растет несоответствие между нашей психологией - психологией, которая эволюционировала для работы с небольшими группами людей в совершенно другой социальной и физической среде, – и всё более технологичным и глобализованным миром, в котором мы живем. Поскольку человеческие взаимодействия всё чаще заменяются системами «человек-компьютер» для покупок, работы и развлечений, мы постепенно всё меньше и меньше получаем обратную реакцию от мозга. Теперь причины и последствия наших решений и действий часто не могут вызвать соответствующего рационального ответа. Проще говоря, мы видим все меньше и меньше результатов наших действий, даже если они оказывают все большее и большее влияние на мир в целом. Так, осведомленность каждого о ошибках, возникающих в процессе мышления, позволит значительно эффективнее принимать ежедневные решения.

**Список использованных источников:**

1. Kahneman, D. Thinking, Fast and Slow/ D. Kahneman. – Penguin Books: New York, NY, USA, 2011. – 499p.
2. Todd, P.M., Gigerenzer, G. Precise of Simple Heuristics that Makes Us Smart/ P.M. Todd, G. Gigerenzer // Behav. Brain Sci. 2000. – 23. – P.727–780.
3. Tversky A. Judgement under uncertainty: Heuristics and biases / A. Tversky, D. Kahneman // Sciences 185, 1974. - P. 1124-1131.

## ТЕОРИЯ И ИСТОРИЯ ТРАНСГУМАНИЗМА

*Мельник Н.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Миськевич В.И. – канд. фил. наук, доцент*

В данной статье освещены основные этапы развития трансгуманизма в качестве философской концепции, современного движения. Рассмотрены различные философско-литературные течения как биокосмизм-иммортизм, русский космизм, а также видение человека будущего их последователями. Было проведено сравнение первоначального и современного значения термина «трансгуманизм». Были рассмотрены основные объединения и организации трансгуманистов по всему миру и их деятельность, влияние на развитие современной науки.

Никакой человек в мире не родится готовым, то есть вполне сформировавшимся, но всякая его жизнь есть не что иное, как непрерывно движущееся развитие, беспрестанное формирование.

В. Г. Белинский

На протяжении многих столетий человек искал различные способы духовного саморазвития и изучения окружающего его мира. Человечество всегда интересовало происхождение нашей вселенной, место человека в ней. Наибольшим достижением мысли эпохи возрождения является формирование идей гуманизма. Исходя из устава Международного гуманистического и этического союза, гуманизм является демократической, этической жизненной позицией, которая утверждает, что люди имеют право и обязанность определять смысл и форму своей жизни. Гуманизм призывает к построению более гуманного общества посредством этики, основанной на таких ценностях, как свобода, разум, ответственность и свободный поиск, на основе использования человеческих способностей. Эти идеи легли в основание мировоззрения современного общества [1].

В конце XIX века - середине XX века активно начинается развиваться техника, общество пересматривает свои устоявшиеся идеалы, возникает кризис гуманизма. Именно в эпоху активного и быстрого технического прогресса как никогда остро стала проблема этики и угроз, связанных с некорректным использованием достижений техники. Для изучения данных, актуальных и по сей день, вопросов человеку нужны были новые философские концепции, теории, движения, способные определить новые ценности и моральные устои общества. Философской концепцией, осмысляющей

все риски внедрения достижений прогресса, утверждающей, что технологии смогут изменить физическое тело человека в ходе эволюции, и описывающей устав нового общества, является трансгуманизм.

Трансгуманизм как философская концепция берет свои истоки из многих философских и литературных течений [2]. Одним из самых первых течений, развивающих идею бессмертия человека, является русский космизм-иммортиализм. Русский космизм был революционным движением для конца XIX – начала XX века. Представители данной концепции, основываясь на факте цефализации, роста головного мозга и усложнении нервной системы, делают вывод о том, что человек - это еще только промежуточная стадия, что вполне возможна эволюция организма человека, его дальнейшее развитие как вида. У различных активистов космизма новая ступень развития человека имеет различные названия: у Федорова - регуляции природы, у Вернадского - ноосферы, у Флоренского - пневмосферы. Но все они сходятся на том, что главной особенностью «нового» человека является преодоление основного закона природы о выживании сильнейшего, расширение власти духа над материей и установление нового принципа жизни, основанного на солидарности и равенстве живых существ, любви и бессмертии.

В середине XX века наука и медицина особо активно прогрессируют, появляется крионика и многие новые направления в технике. Начинается новый этап развития идеи иммортализма. Впервые термин “трансгуманизм” был введен биологом и основателем ЮНЕСКО Джулианом Хаксли в 1957 г. Трансгуманизм Хаксли является концепцией искусственного отбора человечества, которая тесно связана с запрещенной во многих странах псевдонаукой – евгеникой [3]. Отображая антимораль и противоречия гуманистическим принципам общества, данное значение не соответствует общепринятому трансгуманизму. Современные трансгуманисты считают, что такие ученые 1920 - 1950-х годов, как биолог Дж. Б. С. Холдейн, физик Дж. Д. Бернал, а также палеонтолог П. Тейяр де Шарден, являются идейными предшественниками концепции. Однако наибольшее влияние на формирование современного трансгуманизма оказали фантаст и футуролог Ф. М. Эсфендиари, который в 70-х годах прошлого века использовал псевдоним FM-2030, основатель крионики Р. Эттингер, американские ученые К. Э. Дрекслер и М. Л. Минский. Именно с их трудами трансгуманисты связывают наиболее перспективные и важные направления развития науки такие, как крионика, иммортализм, искусственный интеллект и нанотехнологии, которые могут быть использованы для совершенствования человека, достижения им бессмертия. Иммортализм является системой взглядов, главная цель которой - достижение бесконечного долголетия. Искусственный интеллект - программа максимально подобная мышлению человека, его поведению и эмоциональности, способная в будущем заменить самого человека. Нанотехнологии понимаются как область науки и техники, разрабатывающая продукты определенной атомной структуры.

Главным теоретическим базисом для трансгуманизма служит теория о мыслящей материи, разработанная в 2000 году А. В. Мищенко - кандидатом физико-математических наук, ярким активистом и популяризатором движения [4]. Теория была создана для описания последующей эволюции человечества, изучения ее различных вариантов. Автор теории выделяет понятие организованной материи. Организованная материя – это материя, частицы которой более упорядочены, чем у окружающей ее среды, соответственно она обладает меньшей энтропией. Материя разделена на несколько видов: физическая, химическая, биологическая и мыслящая. Каждый из видов соответствует определенному этапу развития живых существ на Земле. Из теории следует, что человек в данный момент состоит из биологической материи (физическая оболочка) и мыслящей материи (сознание). Одной из главных задач трансгуманизма является разделение этих двух составляющих, т.е. в будущем мышление не будет зависеть от физической оболочки. Схожие идеи ранее существовали в публикациях знаменитого изобретателя и последователя космизма К. Э. Циолковского. В его трудах появляется идея об усовершенствовании человеческого тела для дальнейшего беспрепятственного перемещения по космосу, а именно отделения сознания от тела. Что в очередной раз подтверждает идейную связь между трансгуманизмом и русским космизмом-иммортализмом.

В результате продолжительного процесса становления трансгуманизма в конце XX-го века были созданы такие основные для движения сообщества и объединения, как Humanity+ (Дэмиен Бродерик, Макс Мор, Наташа Вита-Мор, Ник Бостром) в США, Российское трансгуманистическое движение (Д. Рязанов, Д. Медведев, В. Косарев, И. Артюхов, И.Судак, И.Кирилук), члены которого являются гражданами стран СНГ. Важными для становления движения стали официальные документы, принятые этими объединениями (Transhumanism word declaration, Манифест Российского Трансгуманистического Движения, Устав МОО “Трансгуманистическое движение”), закрепляющие основные идеи, цели и направления движения. Стоит особо отметить, что трансгуманисты осознают проблему опасности слишком быстрого и неконтролируемого развития технического прогресса, именно поэтому одной из их основных целей является “содействие развитию новых и традиционных наук и технологий в той их части, которая непосредственно касается трансгуманистических и имморталистических аспектов бытия человечества, а также безопасности использования этих

технологий"[5], а одним из главных принципов движения является его отстраненность от политики [6]. Резюмируя вышеизложенное, можно сделать вывод о том, что современный трансгуманизм является квинтэссенцией идей различных историко-философских и литературных течений, оказавших влияние на видение мира современного человека, что также подтверждено идейным основанием теории о мыслящей материи. Благодаря активному развитию сферы новых технологий и внедрению новых устройств в повседневную жизнь, интерес к данной философской концепции возрастает с каждым годом, особенно среди молодых людей, стремящихся приобщиться к философским идеям. Также приверженцами трансгуманизма являются знаменитые политические деятели, крупные бизнесмены и ученые. Благодаря трансгуманизму популяризируются специальности и направления связанные с инженерией и разработкой новых устройств. Основные организации трансгуманистов активно сотрудничают с исследователями и учеными, целью которых является увеличение продолжительности человеческой жизни, как традиционными методами, так и с использованием инновационных технологий (генная инженерия, наномедицина, крионика и др.), что непременно окажет положительный эффект на развитие современной науки и техники.

**Список использованных источников:**

1. Дараган Н. Д. О гуманизме эпохи возрождения / Н. Д. Дараган // Вестник Таганрогского института имени А. П. Чехова, Таганрог 2007..
2. Луков В. А. Трансгуманизм / В. А. Луков // Знание. Понимание. Умение. Москва 2017.
3. Артюхов И. В. Динамика ценностных ориентаций в современной культуре: поиск оптимальности в экстремальных условиях / И. В. Артюхов // М: VII международный философско-культурологический конгресс 2004.
4. Мищенко А. Трансгуманизм и теория мыслящей материи / Алесь Мищенко // Москва 2015.
5. Устав общероссийской общественной организации «Российское трансгуманистическое движение» (РТД) // Смоленск 2005.
6. Transhumanism word declaration. // Boston 1998.

## **СОЗДАНИЕ СТРАТЕГИЙ ПОВЕДЕНИЯ ЧЕЛОВЕКА ПРИ ПОМОЩИ ИНСТРУМЕНТОВ НЛП**

*Папкевич Г.И.*

*Белорусский государственный университет информатики и радиоэлектроники*

*г. Минск, Республика Беларусь*

*Шепетюк В.В. – канд. филос. наук*

Осознанно решая повседневные задачи, человек может даже задуматься о том, каким образом происходит решение этих задач на подсознательном уровне. Для того чтобы облегчить поиск этих "механизмов" мы можем обратиться к выявленным разработчиками НЛП механизмам. За основу они взяли уже разработанную ранее модель ТОТЕ (Test-Operate-Test-Exit). Эта модель описывает блок-схему субъективного опыта поведения человека, начиная от стимула к действию, и заканчивая внутренней «обработкой» информации в терминах проверки человеком внешнего стимула по сравнению с внутренними планами (мыслями), а также действия либо над стимулом, либо над картой для их изменения, позволяющей осуществить проверку конгруэнтности или ее отсутствия для итогового выхода из «программы» поведения. Роберт Дилтс, один из первых разработчиков НЛП, рассказывал об этой модели на примере настройки радио: «Когда вы настраиваете громкость радиоприемника или магнитофона, вы непрерывно проверяете громкость звука, прислушиваясь к нему. Если звук слишком тихий, вы совершаете действие, поворачивая ручку по часовой стрелке. Если вы перестараетесь и звук станет слишком громким, вы поворачиваете ручку против часовой стрелки, уменьшая громкость звука. Когда вы настроите усилитель на подходящую громкость, вы выходите из ТОТЕ «настройка громкости звука», усаживаетесь в удобное кресло и продолжаете чтение». Создатели НЛП улучшили модель ТОТЕ, определив, каким образом мы осуществляем проверку (Test) и действие (Operate) в терминах т.н. репрезентативной системы и субмодальностей. Например, человек может сравнить то, как он помнит правописание определённого слова (Визуальная внутренняя (Vi) репрезентация) и то, как оно записано в словаре (Визуальная внешняя (Ve) репрезентация). Человек мог сделать это и в кинестетической (Ki/Ke) или в аудиальной (Ai/Ae) системе. Если звук, интенсивность, размер, цвет или образ достигли определённого порогового значения, то достигается конгруэнтность, равновесие. Конгруэнтность, таким образом, приводит к «выходу из программы», а неконгруэнтность приводит к «продолжению программы», которые также представлены с использованием одной из репрезентативных систем. Чтобы «распаковать» какую-либо стратегию, нам нужно сначала «проговорить» какую-либо уже решенную проблему. Возьмем незамысловатый и в тоже время типичный пример всеми нелюбимого утреннего будильника. Порядок действий здесь будет следующим. Я слышу звонок будильника, затем смотрю на него, когда



выключаю. Затем я снова ложусь и чувствую насколько удобно в кровати. Но через мгновение в голове может пронестись образ того, как я опаздываю на важную встречу. Этот образ не покидает меня и постепенно усиливается все сильнее и сильнее. И когда неприятные образы становятся достаточно сильными, я встаю. В репрезентативных терминах данную стратегию можно записать следующим образом:

$Ae \rightarrow Ve \rightarrow Ke \rightarrow Vi/Ai/Ki \rightarrow \text{Выход (Exit)}$

где Ae – аудиальный внешний источник информации; Ve – визуальный внешний источник информации; Ke – кинестетический внешний источник информации; Vi/Ai/Ki – визуальный/аудиальный/кинестетический внутренний источник информации.

Для изменения неэффективных стратегий человека, которые всегда находится в состоянии «даунтайм», мы встраиваем в них большее количество внешних проверок (Test). Так, для человека, боящегося публичных выступлений, мы встраиваем в стратегию этап, на котором он получает доступ к состоянию расслабленности, комфорта, самоуверенности. При разработке стратегий мы встраиваем т.н. «якоря» контекста и точки зрения принятия решения (Operate), которые управляют нейролингвистическими процессами. Такая разработка стратегий нужна для того, чтобы чувствовать себя комфортно при разных контекстах. Когда мы используем стратегию для моделирования, мы находим человека, который уже обладает способностью достигать результата, и просто определяем либо используем его стратегию в качестве модели.

Для встраивания стратегий в свою повседневную жизнь существует несколько приемов. Давайте рассмотрим один из них – якорение. Эта методика подразумевает якорение репрезентации или состояния и введение в стратегию этапов при повторении человеком последовательности действий (эмпирический путь). Когда мы осуществляем якорение последовательности, мы должны прикреплять ее к некоторому контекстуальному стимулу. Извлеките этапы стратегии при помощи наводящих вопросов и наблюдения и осуществите затем якорение каждого этапа при помощи одного и того же якоря. Например, мы хотим получить доступ к стратегии мотивации. Наводящим вопросом в данном примере будет: «Вспомните, когда в последний раз вы чувствовали себя по-настоящему мотивированным». Далее нужно осуществить якорение данного опыта. Позднее мы можем воспроизвести данный якорь в другом контексте, так чтобы человек заново получил доступ к этой последовательности стратегии мотивации. Паттерны синестезии автоматически поддерживают собственные, однажды инициированные процессы человека. Например, человек с боязнью высоты может посмотреть вниз с крыши дома и немедленно испытать фобическую реакцию. Он видит (V) высоту и чувствует (K) страх. Синестезия состоит из двух компонентов репрезентативных систем, при этом первая репрезентативная система немедленно запускает вторую. Мы осуществляем якорение этой синестезии и затем переносим ее в другие ситуации. Это может ускорить чрезмерно долгую стратегию и позволит избежать бесконечных тупиковых циклов при встрече с этой фобией вновь.

Рассмотрим ещё одну типичную проблему – прокрастинацию. Например, у одного человека была громоздкая и неэффективная стратегия принятия решений. Он проводил много дней в размышлениях и откладывал решение, пока не упускал основные возможности решения. После этого он злился на себя за то, что потерял такое большое количество времени. При разработке стратегии в данном случае, он рассматривал в первую очередь «вероятность упустить благоприятные возможности и потерять время» (Ad/K). Вопрос «Как вы себя чувствуете (K) в конце вашей стратегии, когда осознаете (Ad/V/A/K) ценность потерянного времени и упущенных возможностей» помогает построить новую стратегию. Это ускорит новый порядок действий и послужит необходимой мотивацией для своевременного принятия решений. Он воспримет это как ресурс при проверке своего календарного плана и использует свои отрицательные чувства (как и в примере с будильником) в качестве основания для принятия другого решения.

В заключении отметим, что у индивида часто может появиться состояние «ступора» и полное непонимание того, как справляться с возникающими жизненными задачами и трудностями. Однако если он начнет осознавать то, что большинство его задач и трудностей построены на типичных стратегиях поведения, то он сможет преодолевать их более эффективно. Также следует помнить о том, что грамотное осознанное использование репрезентативных систем поможет в понимании и управлении основными принципами реализации стратегий поведения.

**Список использованных источников:**

1. The Complete Manual for Neuro-Linguistic Programming Practitioner Certification: Volume 1. Mastering Systemic NLP: Volume 2, 2019. – P. 233-258.
2. «Level IV» in Neuro-Linguistic Programming: Skills for Communication and Change (Reynoldsburg: L.E.A.D. Consultants, Inc.), 1986. - P. 6–19.
3. Nlp Going Meta: Advanced Modeling Using Meta-Levels, 1997. - P. 158-184.

## РАССУЖДЕНИЯ О СМЕРТИ И СТАРЕНИИ

*Рафеев П.Ю.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Лагунова Е.Н. – преподаватель*

О смерти в своё время задумывался каждый из нас. Кто-то считает её несправедливой, кто-то соглашается с её необходимостью. И, как ни парадоксально это замечать, и те, и другие правы: просто некоторые виды смерти более не нужны, а иные по-прежнему актуальны. Наиболее же неоднозначным остаётся вопрос необходимости смерти от старости и старения в целом, тем более с учётом новейших микробиологических разработок, способных если не избавить человечество от старения полностью, то по крайней мере существенно увеличить максимально возможную продолжительность жизни.

Испокон веков (с тех пор, как впервые осознали неминуемость смерти, если быть точным) люди боялись смерти. Под влиянием этого страха они научились лечить огромное множество заболеваний, создали законодательства, предусматривающие жестокую кару за убийство, разработали технику безопасного поведения в различных ситуациях, даже научились делать сложнейшие хирургические операции. Они считали смерть несправедливой, утверждая, что она не позволяет достичь смысла их жизни и исполнить то, зачем они родились. Впоследствии они даже придумали загробный мир, жизнь после смерти, только чтобы не смиряться с неминуемой перспективой прекратить мыслить, а значит и существовать.

Но если смерть так ужасна, встаёт резонный вопрос: зачем она нужна. Природа ведь ничего не создаёт просто так. За почти 4 миллиарда лет она, посредством естественного отбора, создала очень сложную, логически взаимосвязанную, устойчивую и развивающуюся систему под названием жизнь (то есть совокупность всех живых существ), а потому всё, что она сотворила, имеет важную причину. Для понимания причин необходимости смерти можно мысленно представить себе мир, в котором смерть была побеждена.

На ум оптимистам сразу приходит радужная картина всеобщего счастья, вечной жизни учёных, творцов, великих правителей, у которых впереди была бы целая вечность дабы улучшать наш мир, делать его прекраснее, удобнее, лучше.

А вот пессимисты сразу замечают истинную, обратную сторону медали: если смерть исчезнет, перенаселение наступит за считанные годы, большинство людей будут вечно жить и страдать в голоде и нищете, к тому же лишённые какой-либо мотивации менять свою жизнь (ибо с исчезновением смерти базовые инстинкты, вроде инстинкта самосохранения, также исчезнут). Вдобавок, если исчезнет смерть – исчезнет и естественный отбор. Эволюция остановится, и жизнь окончательно потеряет смысл (если она когда-то вообще его имела). Перспектива явно не из приятных.

С другой стороны, борьба со смертью как с чем-то неправильным позволила человечеству достичь немалых успехов, увеличив среднюю продолжительность жизни с 20 лет почти до 80. Безусловно, это привело к невероятному росту численности людского населения (близкой к перенаселению планеты), однако нельзя отрицать, что без увеличения продолжительности жизни нынешний уровень развития цивилизации был бы недостижим, ибо почти все великие люди меняли этот мир будучи старше 20-ти лет.

Эти противоречия между ожидаемым вредом от борьбы со смертью и наблюдаемыми результатами разрешаются, если представить, что человечество боролось не со смертью в целом, а лишь с отдельными её проявлениями: убийствами, приобретёнными заболеваниями и старением.

При этом остальные проявления смерти, такие как голод, врождённые заболевания, стихийные бедствия, суицид и другие, должны оставаться, и сейчас вы поймёте, почему. Для этого следует разобраться, что вызвало у природы необходимость в сотворении тех или иных проявлений смерти:

1. Убийства появились для выявления наиболее сильных и выносливых представителей вида, которые зачастую выживали и убивали более слабых представителей вида, тем самым создавая ещё один путь для осуществления естественного отбора. Так было до появления человека и во времена первобытного общества, а что же сейчас? Сейчас убийцами становятся не наиболее сильные или выносливые представители рода людского, а наиболее хитрые, подлые, мстительные, жадные и ещё бог весть какие, но только не наилучшие.

Таким образом, убийство, как проявление естественного отбора, ускоряющее и направляющее эволюцию, потеряло свои функции и свой смысл, а потому должно быть искоренено.

2. Приобретённые заболевания – яркий пример межвидовой борьбы (между людьми и болезнетворными микроорганизмами). Изначально единственным оружием людей в этой схватке был иммунитет, и сама схватка была призвана развивать его, дабы впоследствии использовать сильный иммунитет не только против внешнего болезнетворного врага, но и против собственных мутировавших клеток, вызывающих рак и иные онкологические заболевания.

Со временем же у человечества появился гораздо более эффективный инструмент борьбы с заболеваниями: развитая медицина. А так как медицина развивается гораздо быстрее иммунитета, логично сделать ставку именно на неё, что делает смерть от приобретённых заболеваний, призванную развивать иммунитет, бессмысленной. Значит она должна быть искоренена.

3. Голод – один из наилучших естественных способов контролировать численность населения. Всё просто: чем больше людей, и чем менее они приспособлены себя прокармливать, тем менее они достойны жить (с точки зрения природы, разумеется). Так как на сегодняшний день перенаселение является одной из основных угроз, а в странах с наибольшим приростом населения (африканских и восточноазиатских странах) лимитирующим фактором является именно голод, его актуальность очевидна.

Голод как проявление смерти в современном мире всё ещё необходим.

4. С врождёнными заболеваниями всё просто. В естественных условиях они привели бы к смерти своего носителя как более слабого представителя вида с самого момента рождения, тем самым предотвратив передачу заболевания или мутации по наследству. А значит, в современных реалиях позволение жить человеку с врождёнными заболеваниями равносильно заражению его потомства этими же заболеваниями, причём даже не спрашивая это самое “потомство”.

Мы не должны противопоставлять себя природе и культивировать эти заболевания, ослабляющие наш вид. Нужно позволять их носителям умирать, таким образом смерть от них – это естественный процесс, которому мы не должны препятствовать.

5. Стихийные бедствия в естественных условиях достаточно редки, однако с нарастанием техногенной катастрофы они, естественно, будут случаться чаще, унося всё больше жизней. И, раз уж человечество хочет снизить смертность от стихийных бедствий, следует бороться не со следствием (то есть с самими бедствиями), а с причиной: негативным влиянием человека на окружающую среду. И до тех пор, пока научно-технический прогресс не войдёт в гармонию с природой, смерть от стихийных бедствий будет для человечества сигналом о существующих проблемах и необходимости их решения.

А значит, смерть от стихийных бедствий пока является необходимостью и должна быть принимаема во внимание.

6. Суицид – это одновременно и способ естественного отбора отсеять наиболее психологически слабых людей, и способ человеку самому выбирать свою судьбу. Кроме того, бывают ситуации, когда не само самоубийство, а именно мысли о нём позволяют пережить самые трудные испытания в жизни. Ведь если можно умереть в любой момент по своему желанию, то почему бы не попробовать сделать что-нибудь (даже если станет хуже – как минимум один выход будет всегда).

Если бы же суицида не было, подобные мысли не могли бы нам помочь в трудную минуту. А значит, суицид по-прежнему необходим и актуален как одно из проявлений смерти.

7. И, наконец, старение. Смерть от старости стала распространённой совсем недавно (когда значительная часть людей стала доживать до этой самой старости). Старение же как процесс появился у первых многоклеточных животных, вызвав резкое ускорение эволюции среди них. Другой же целью старения стало ограничение численности населения. Однако сейчас, когда развитие генной инженерии достигло того уровня, что человечество может само улучшать свои гены без необходимости эволюции, а рождаемость в большинстве не голодающих регионов планеты и так ниже смертности, старение потеряло свой изначальный, заложенный эволюцией смысл.

Учёные-микробиологи уже знают, как избавить человечество от этого главного пережитка прошлого, ведь после отмены старения наиболее здравомыслящие, здоровые и приспособленные люди смогут жить почти неограниченно долго, в то время как слабые, глупые и неприспособленные будут умирать, предотвращая перенаселение и не мучая в первую очередь себя своим полным страданий бессмысленным существованием.

Итак, подведём итог. Человечеству следует бороться не со смертью в целом, а с отдельными её проявлениями. Убийства и приобретённые заболевания уже давно не считаются нормой, что правильно. Голод, врождённые заболевания, стихийные бедствия и суицид пока ещё необходимы нам для предотвращения куда больших угроз, чем они сами (не исключено, однако, что со временем надобность и в них отпадёт также, но случится это не в ближайшем будущем).

Со старением же всерьёз бороться ещё не начали, хотя, как я пояснил выше, борьба с ним должна считаться одной из первостепенных задач на данный момент. Ведь когда-нибудь в далёком будущем, когда жизнь сможет быть практически вечной, люди не будут считать старение обыденным порядком вещей.

А если мы знаем, как они будут думать, то почему бы нам не думать так же уже сейчас?

**Список использованных источников:**

1. Веселкин П. Н. *Болезнь / П.Н. Веселкин // Большая медицинская энциклопедия гл. ред Б. В. Петровский. — 3 изд. — Москва: Советская энциклопедия, 1976 — с. 86.*
2. Крамер С. *50 дней до моей смерти / С. Крамер. — Москва: АСТ, 2015. — 196-201 с.*
3. David, Grigg. *The historiography of hunger: changing views on the world food problem 1945-1980 / David Grigg // Transactions of the Institute of British Geographers, 1981. – Vol. 6. – No. 3. – P. 279-292.*

## ПРОБЛЕМЫ ФЕМИНИЗМА В СОВРЕМЕННОМ ОБЩЕСТВЕ

*Хиневич Ю.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Миськевич В.И. – канд. фил. наук, доцент*

Она кричала: "Равноправие!  
Отдай мне все мои права!"  
Он тихо: «Да бери. На здоровье.  
Все отдаю, ведь ты права.  
Водить машину? Да пожалуйста.  
Белить, стирать, пилить, строгать.  
Все отдаю, потом не жалуйся  
Не смей канючить и кричать.  
С азартом, женщина, удвоенным  
Взялась сама за все дела.  
Стирала, дом построить — строила.  
Рулила, рыла и гребла...  
А он, прочувствовав «бесправие»,  
Облюбовав себе диван,  
Благодарит, за равноправие,  
Всех женщин мира и всех стран...

Феминизм уже стал свойственной частью современного общества. Только представьте себе мир, где женщины не могут голосовать и занимать руководящие должности, не имеют права на получение высшего образования, не должны носить брюки, короткие юбки и краситься- это удел "падших" женщин. Трудно представить? А ведь совсем недавно всё так и было. К слову, в цивилизованной Швейцарии женщинам было разрешено голосовать на выборах только в 80-е годы XX века. Поэтому нельзя говорить о том, что феминизм утратил свою актуальность. А по поводу того, что тяжело делать все самой – но ведь и раньше женщин вовсе не на руках носили, да еще и слова сказать не давали. Женщиной всегда было трудно быть. Они многого добились, но что мы видим? Что сейчас женщина ставит в приоритеты карьеру и самоутверждение на первое место, а обязанности матери и жены сдвигаются на задний план. Как-то так получилось, что покричать женщины покричали, а истинную свободу от мужского шовинизма так и не приобрели.

При этом в современном обществе сложилось огромное количество мифов о деятельности феминисток. Знаете ли вы, что феминистки никогда не устраивали публичных сожжений лифчиков? В 1968 году американские студентки протестовали против конкурса красоты «Мисс Америка». Они устроили коронацию овцы и демонстративно выкидывали дамские журналы, туфли на каблуках, бигуди и корсеты. Так же студентки собирались заодно и жечь нижнее белье, но делать этого не стали. Только очень уж понравился редактору New York Post заголовок «Сжигательницы лифчиков» — и правда, звучит романтично и устрашающе. Так с помощью СМИ и родилось поверие о прожигательных наклонностях феминисток.

«Феминизм — это когда бабы хотят править миром», — сказал однажды мой товарищ. И тем самым сформулировал основной стереотип о феминизме. Феминизм— это движение за права женщин, против их дискриминации по половому признаку. Это значит, что у женщин не должно быть меньше политических, экономических и гражданских прав просто потому, что они женщины; а еще их права должны соблюдаться на практике.

Ранний феминизм появился после Великой Французской Революции. Один из самых ярких персонажей эпохи была Олимпия де Гуж. Она писала в своей «Декларации прав женщины и гражданки»: «Если женщина имеет право взойти на эшафот, то она должна иметь право взойти и на трибуну». Ее казнили в ноябре 1793 года, и она была далеко не единственной. А на трибуну французская революция женщин так и не пустила. Позднее пришедший к власти Наполеон и закрепил в конституции, что гражданскими правами могут обладать только мужчины.

Абигейл Смит Адамс, жена президента Джона Адамса, которую считают первой американской феминисткой, заявила: «Мы не станем подчиняться законам, в принятии которых мы не участвовали, и власти, которая не представляет наших интересов». Из-за этого развивался суфражизм, то есть движение за равные избирательные права.

Основные проблемы, которые поднимают и пытаются решить феминистки во многих странах— это домашнее насилие, которое не все считают преступлением. По статистике 70% убитых женщин убиты их сексуальными партнёрами. И это — лишь то, что отражено в статистике, а ведь лишь некоторые

избитые и изнасилованные женщины сообщают об этом.

В 1736 году английским судьей сэром Мэтью Хейлом было принято решение, на два с половиной столетия определившее юридическую судьбу понятия «изнасилование в браке»: «Муж не может быть виновен в насилии, совершенном им над его законной женой, ибо по их согласию и брачному договору жена отдана своему мужу и не может ни в чем отказывать ему». В 1991 году Апелляционный суд Великобритании постановил, что этот принцип более не соответствует требованиям эпохи, и оставил в силе обвинительный приговор человеку, осужденному за изнасилование жены.

Многие считают, что феминистки мужененавистницы. Но, не смотря на это, статистика доказывает, что представление, будто феминизм — удел людей с неудавшейся личной жизнью и что со счастливыми романтическими гетеросексуальными отношениями он несовместим, — миф. Об этом говорят результаты, полученные специалистами Университета Рутгерса в Нью-Джерси, США, которые провели очное анкетирование 242 студентов и онлайн-опрос 289 человек постарше. Мужчины, поддерживающие отношения с женщинами-феминистками, сообщали исследователям о более стабильных отношениях и большем сексуальном удовлетворении.

А мужененавистничество феминизму приписывают благодаря «стараниям» лишь одного из его многочисленных направлений — радикального феминизма. Ключевым понятием для радикального феминизма является патриархат — система социальных структур и практик, которая подчиняет и угнетает женщин, пронизывая все сферы человеческих отношений. Радикальные феминистки считают причиной угнетения женщин патриархальные гендерные отношения.

Радикальные феминистки известны благодаря своему активизму. Они говорят о необходимости существования исключительно женских движений и настаивают на том, что все мужчины заинтересованы в подавлении женщин и воспроизводят механизмы патриархата.

Я глубоко убеждена, что принципиальная непримиримость яростных сторонниц этих идей возникает из-за их неспособности понять важность роли женщины как женщины и разумно разграничить сферы деятельности женщин и мужчин. Но всё-таки предназначение прекрасной половины человечества заключено в украшении нашей планеты. Женщины должны понять, что они созданы не для того, чтобы прислуживать мужчинам и детям, а, в первую очередь для того, чтобы реализовать себя как талантливую и уникальную личность. А когда мужчины замечает эту личность, то ему в голову не придёт записать вас в прислугу.

Не стоит забывать о женском достоинстве даже в самых трудных житейских ситуациях и помните, что именно тогда мужчины более всего нуждаются в вашей моральной поддержке, а получив ее от вас, они, в конце концов, сумеют справиться с любой бедой на радость и самим себе, и вам, своим любимым.

Хочу закончить отрывком из стихотворения Александра Груздева «Любите женщину за грех..»

Любите женщину за грех, который вынесла из рая,  
А не за то, что лучше всех она готовит и стирает.  
Любите женщину за грусть, которую от вас скрывает.  
За то, что рядом с нею груз проблем быстрее убывает.  
Любите женщину за ум, который и велик и скромн.  
За детского веселья шум рассветным утром в вашем доме.  
Любите женщину за ночь, которую она вам дарит,  
И за желание помочь, когда смертельно вы устали.  
Любите в женщине мечту и интригующую тайну  
Не унижайте красоту упрёком, брошенным случайно.

**Список использованных источников:**

1. Феминизм: Восток, Запад, Россия. М., 1993.;
2. Успенская В. Стереотипы на пути женщин к лидерству // Феминистская теория и практика Восток-Запад. СПб., 1996.
3. Дойл. Р. Мужской манифест. Здравый подход к проблемам пола. Философия и политика Мужского Движения. М. 2005.

## ЛЕЙБНИЦ И ОСНОВАНИЯ СИМВОЛИЧЕСКОЙ ЛОГИКИ

*Христенко А. В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Дисько-Шуман М. Р. – кандидат философских наук, доцент*

В работе описана роль исследований Лейбница в рамках науки Нового времени. Рассмотрены основные идеи символической логики, начало которым было положено в трудах Лейбница. Сделан акцент на заслугах Лейбница в развитии логических идей.

1. Проблематика математизации формальной логики

Острая необходимость в математизации логики встала перед учеными Нового времени. Уже тогда высказывались идеи о том, что классическая силлогистика не в состоянии полностью охватить все те выводы, которые производились в результате научных исследований. Нужна была система символов и знаков для формализации языка науки. Однако, человеческий язык слишком многогранен по своей сути. Задача перевода речи человека в систему уравнений до сих пор остается непосильной.

2. Предшественники Лейбница и их попытки

Впервые проблематика создания логики для науки была сформулирована Декартом и Паскалем. Следует отметить идеи Гоббса, утверждавшего, что мышление есть исчисление. Рассмотрения стоит труд последователей Декарта Арно и Николя «Логика Пор-Рояля». Наибольшее влияние на методологию Лейбница оказали: Луллий, желающий механизировать процесс умозаключения, Джордано Бруно, с его концепцией познания, и Декарт, мыслящий о создании универсального логико-математического метода решения научных задач.

3. Лейбниц о «создании мирового языка»

Разрабатываемая Лейбницем логическая теория, которую он сам называл «всеобщей характеристикой», содержала в себе идеал создать «мировой язык», что будет понятен всем и каждому. Там же было выдвинуто требование выработать универсальный логический язык и символику. Лейбниц желал добиться результата, при котором было бы возможно «посчитать» истину, исходя из реплик дискутирующих. По сути, встала задача алгоритмизации творческого мышления человека, над чем философ был готов трудиться сколь угодно долго. «Логика должна стать универсальной математикой, а не развлечением для школьников», - говорил он.

4. Понятие истины и логические законы логики вводимые Лейбницем

В основу своего научного познания Лейбниц ставит понятие истины, а именно истинность суждения. Он говорит, что важно строить заключение не на количестве наблюдающихся случаев, а на их качестве. На вооружение в выстраивании выводов Лейбниц берет законы логики, которые формулирует по-своему. Особое внимание уделяет закону тождества и закону достаточного основания. Доказывает, что они лежат в основе силлогистики и всякого выводного знания вообще.

5. Основные этапы формирования символической логики

Выделяют три этапа в построении логического исчисления Лейбницем. В первом он пытается каждому понятию придать число и связать их в верном равенстве арифметики, но данный подход оказался слишком трудоемким в подборе чисел и в некоторых случаях недостаточно эффективным. Во втором этапе Лейбниц вводит символы связи, логические константы, операцию логического умножения, но в целом сохраняет ряд словесных связей. Третий этап расширения своей логической системы связан с попыткой определить понятие «содержаться в», вводится операция логического вычитания и другие бинарные отношения. Следует отметить вклад Лейбница в разработку некоторых вопросов традиционной логики и тот факт, что он пользовался кругами Эйлера еще до самого Эйлера.

6. Значение трудов Лейбница

Лейбниц не довел свое логическое исчисление до совершенства, а идеи «посчитать» истину и создать свою пазиграфию так и остались утопией, нельзя отрицать значение развиваемых им теоретических и практических концепций. Символика Лейбница была в должной мере оценена и отшлифована будущими учеными, позволив им получить совершенный инструмент построения научных выводов. Не раз верно утверждалось, что «Лейбниц не только математизирует логику, но и логизирует математику»[1, с. 221]. Его труды легли в основу современных логических исчислений, в связи с чем, Лейбница по праву можно считать прародителем символической логики.

**Список использованных источников:**

1. Стяжкин, Н. И. Формирование математической логики: Москва: Изд-во «Наука», 1967. – 198 с.
2. Маковельский, А. О. История логики: Изд-во «Кучково поле», 2004.

**СЕКЦИЯ «ФИЛОСОФИЯ И МЕТОДОЛОГИЯ НАУКИ»**

## «ВОЛЯ К ЖИЗНИ» АРТУРА ШОПЕНГАУЭРА, ИЛИ ПРОЛЕГОМЕНЫ «ВОЛИ К ВЛАСТИ» ФРИДРИХА НИЦШЕ

Самусёв И. С.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Антипов В. И. – преподаватель

В работе определяются особенности иррационального мотива в онтологии человеческого бытия с точки зрения понятий «Воли к жизни», «Воли к власти». Рассматривается процесс формулирования и объяснения понимания ницшеанской «Воли к власти» через онтологию воли Шопенгауэра. Научная новизна заключается в рассмотрении допущения иррациональной «воли к жизни» как исходных понятий для ницшеанской иррациональности. Сделан вывод о актуальности данной проблематики в современных реалиях.

В настоящее время, рациональный логос, охватывающий, казалось бы, все сферы человеческого знания, начиная от строгой научной формализации, заканчивая неререфлексивной повседневной области человеческого бытия, мы сталкиваемся с проблемой внесения иррациональных аспектов существования человека, в рациональный, научный логос. Причем, не как имеющая место быть ошибка, недостаток, исключение, аномалия, заблуждение и т.д., а как его важнейший элемент, формирующий в нем онтологическое единство. Во всяком случае, несмотря на сегодняшние тенденции привести в современный мир гуманизм, процветание и мирную обстановку, мы постоянно можем наблюдать проявления человеческой «воли к жизни» или «воли к власти» на самых различных уровнях общественной жизни: как в экзистенциально-личностном, социальном взаимодействии, политическом взаимодействии между тем или иным обществом. Один из важнейших тезисов при наблюдении за данными феноменами общественного взаимодействия и личностного отношения самого к себе, что так или иначе, мы можем наблюдать некоторый **иррациональный** мотив присущий человеку, который онтологически врос в его бытие. Данное заключение, дает нам дополнительное основание в необходимости и актуальности, вновь и вновь возвращаться, переосмыслять, интерпретировать, использовать труды, родоначальников иррациональной философии Фридриха Ницше и Артура Шопенгауэра.

«Мир как воля и представление» - это центральная работа немецкого философа Артура Шопенгауэра, которую он назвал «самой серьезной и самой важной из наших книг». А. Шопенгауэр использовал слово «воля» как наиболее известное указание на концепцию, которую можно обозначить также словами «вожделение», «стремление», «желание», «усилие», «призыв». Философия А. Шопенгауэра придерживается того, что вся природа, включая человека, - это выражение ненасытной «воли к жизни». Именно благодаря этому «желанию жить» человечество испытывает страдание. А желание большего в свою очередь является причиной ещё больших страданий. «Воля к жизни» реализуется в человеке путем эгоизма. Эгоизм в данном случае рассматривается как утверждение собственного существования. Но при этом воля проявляет себя двояко. С одной стороны, она является источником безудержного эгоизма, а с другой - реализует себя в свободе. И поэтому человек может пойти против эгоизма по дороге самопожертвования. Для этого нужно относиться к другому, как к самому себе, и обогащать свой духовный мир, но, по мнению философа, это удел только для редких и избранных. А. Шопенгауэр: «И мы видим очень много людей... трудолюбивых... занятых приумножением богатства». Можно сделать вывод, что у А. Шопенгауэра воля - это «вещь в себе». Только воля способна определить все сущее и влиять на него. Воля - это высший космический принцип, который лежит в основе мироздания. Воля - это «воля к жизни», стремление к ней. Данный взгляд на человеческую волю напоминает нам не менее известную идею, сформированную Фридрихом Ницше уже после смерти Артура Шопенгауэра и названную «воля к власти».

Если человек, который лучше всех понимал Карла Маркса был Фридрих Энгельс. То суждение, что лучше всех понимал А. Шопенгауэра был Ф. Ницше не покажется нам столь амбициозным. «Воля к власти» - это одна из разновидностей волевых импульсов человеческого поведения. «Волю к власти» Ф. Ницше считал определяющим стимулом деятельности и главной способностью человека. Основой жизни, по Ницше, является «воля к власти» или тяга всего живого к самоутверждению, всемогуществу, стремлению расширить власть. На протяжении всей жизни человек стремится достичь максимума чувства власти.

Связь «воли к власти» и «воли к жизни» нельзя назвать однозначно очевидной. Мы можем найти как сходства, так и различия двух философских взглядов. Однако, один из важнейших тезисов для нас, что «воля к власти» не является неким логическим продолжением «воли к жизни», но что крайне важно, «воля к жизни» выступает как некоторый фундамент, можно даже сказать введение в область иррационального волевого мотива человека, на котором выстраивается уже крайне тяжело структурированная надстройка в виде «воли к власти». Если «воля к жизни» рассматривалась как



единственное свойство мироздания, само мироздание то, в свою очередь, воля к власти описывает то, что Ницше считал главной движущей силой в людях: достижения, амбиции и стремление достичь максимально возможного положения в жизни.

Таким образом, мы не ставили цель объединить «волю к жизни» и «волю к власти» в одну систему в котором её элементы будут находиться в иерархической структуре. Особое значение имеет необходимость рассматривать метафизическую волю А. Шопенгауэра, её онтологические элементы, как пролегомены к эгоистичной «воли к власти» Ф. Ницше.

**Список использованных источников:**

1. Ницше, Ф. Воля к власти. Опыт переоценки всех ценностей / Пер. с нем. Е.Герцык и др.-М.:Культурная революция, 2005.
2. Шопенгауэр, А. Собрание сочинений в пяти томах. Том первый. / Перевод Ю. И. Айхенвальда под ред. Ю. Н. Попова. М., «Московский клуб», 1992
3. Шопенгауэр, А. Мир как воля и представление. Дополнение к первому тому. Дополнения к 4й книге./ А. Шопенгауэр М., 1993

## **ЭСТЕТИЧЕСКИЕ ПАРАДИГМЫ В УСЛОВИЯХ ИНФОРМАЦИОННОГО ОБЩЕСТВА**

*Семенцова К. Д.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Антипов В. И. – преподаватель*

В работе определяются особенности информационного общества и его эстетической парадигмы. Рассматривается возможный процесс переход к информационному типу общества в области эстетики. Научная новизна заключается в рассмотрении актуальности относительно новых способов сотворения и демонстраций произведений искусства и общего языка эстетики. Сделан вывод о актуальности данной проблематики в современных реалиях.

Несмотря на то, что эстетическая проблематика входила в контекст философского знания со времени возникновения философии как таковой, оформление самостоятельного статуса эстетики и закрепление этого названия происходит достаточно поздно. Эстетика по праву носит название философской науки, так как имеет явно выраженный смыслообразующий характер. Философская рациональность, включает в себе, как и строгое научное познание, требующее жесткой верификации на эмпирическое соответствие, так и этическое отношение к реальности, требующее рационально сформулированных норм и запретов, однако, нравственное отношение так или иначе, формируется через отношение двух субъектов выбора т.е. это область межсубъектных отношений. Однако, выделяется третья, кардинально отличающееся область отношения к миру, в виде эстетического отношения, и в первую очередь, основывается на индивидуально-конкретных характеристиках бытийности субъекта отношения. Поскольку эстетические ценности создаются преимущественно в рамках искусства, эстетика может рассматриваться прежде всего, как наука о своеобразии искусства и художественного творчества. Искусство оказывает решающее влияние на развитие эстетики. Преследуя цель, понять, что представляет собой эстетика информационного общества, какова ее парадигма и основные черты, выявим особенности понятия «парадигмы» в эстетике, а также некоторые особенности бытия человека «homo informaticus», существование которого имеет некоторые отличия от привычного человека аграрного, индустриального, постиндустриального общества.

Эстетическая парадигма, понятие условное, т.к. каждый отдельный период эстетики, формируется на основе уникальных характеристик человека и социума которое его окружает, формируя в нем определенную субъективность. Однако, парадигма — это попытка найти наиболее общий фундамент, формирующий некоторый диктат в эстетическом восприятии, которое в свою очередь, транслируется через области искусства, будь то живопись, музыка, поэзия и т.д. Мы выделяем три основные парадигмы в эстетике, это классическая, модернистская и постмодернистская. Формирование модернистской парадигмы, для нас связано с разрывом старой моделью субъекта, субъекта рефлексировующего, что вызвало формирование нового языка искусства, который уже не «копировал реальность», а его самоцель была – уничтожение старых норм форм языка. Постмодернистская парадигма, не переняла основу для себя в виде уничтожения языка, а сформировала новое направление, в виде некоторого преобразования старого языка с помощью иронии.

В сравнении с предшествующими эстетическими парадигмами, такими как классическая, модернистская и постмодернистская, информационная парадигма знаменует начало новой эпохи в

искусстве. Особое значение для этой волны, имеет полностью свободное от социальных институтов трансляция «самой себя» субъекту. Находясь в свободном доступе и не требуя особых условий, а иногда и материальных ресурсов, искусство может принимать абсолютно непредсказуемые формы, открывая путь к самовыражению.

Такие понятия как галерея, музей, выставка, паноптикум, сцена и т.д. уже сейчас имеют тенденцию некоторой трансформации, в следствии чего, как и объём так и содержание данных понятий, будут иметь новый вид. В настоящее время, мы можем наблюдать формирования полностью свободных, информационных технологических платформ, для трансляции художественных произведений, которые нивелируют преимущества, не связанные с их эстетической областью, т.е. уравнивают абсолютно всех авторов, и не могут придавать тому или иному произведению некоторый приоритет, для трансляции человеку. Важно отметить, что тезис о трансформации социальных институтов позволяющим функционировать искусству как таковому, не подразумевает «устаревание» предшествующих норм. Наш тезис подразумевает уничтожение нормы как таковой, понятие парадигмы, как некоторой модели для эстетического творчества и её последующего восприятия будет неприменимо к эстетике «homo informaticus».

Таким образом, наступление эпохи информационного общества, уже сейчас иллюстрирует нам формирование новых принципов бытия человека, его отношения к окружающему миру, его субъективности. Этот процесс, можно наблюдать во всех сферах человеческого существования, однако, хороший успех мы можем наблюдать в искусстве и эстетическом восприятии как таковом.

**Список использованных источников:**

1. Учебно-методический комплекс по курсу «эстетика» / И. М. Наливайко / Эстетика. Учебно-методический комплекс. – Мн.: БГУ, 2001. – 109 с.
2. Тоффлер Э. Третья волна / Э. Тоффлер. — М., 2002.
3. Барт Р. Избранные работы: Семиотика: Поэтика: Пер. с фр. / Сост., общ. ред. и вступ. ст. Г. К. Косикова.— М.: Прогресс, 1989. - С.410-424.

## **СЕКЦИЯ «ФИЗИКА»**

## ОБРАБОТКА ИЗОБРАЖЕНИЙ АТОМНО-СИЛОВОГО МИКРОСКОПА С ПОМОЩЬЮ МЕТОДОВ МАТЕМАТИЧЕСКОЙ СТАТИСТИКИ

Деркач А.В., Карымов А.Г., Посудневский И.И.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Ташлыкова-Бушкевич И.И. – канд. физ.-мат. наук, доцент

В данной работе рассмотрены результаты исследования топографии поверхностей быстрозатвердевших фольг сплавов Al-Cr с различной концентрацией легирующего элемента (хрома), подвергнутых термической обработке. Полученные с помощью атомного силового микроскопа данные, обработанные используя методы математической статистики, демонстрируют зависимость отношения высоты неровностей профиля к среднему значению шага между выступами, а так же значения средней арифметической величины шероховатости от способа обработки поверхности фольги и количества примесей в нее.

Технологии анализа структуры материалов в наше время интенсивно развиваются. Атомно-силовая микроскопия (АСМ), наряду с другими методами исследования поверхностей, такими как растровая электронная микроскопия или сканирующая туннельная микроскопия, широко применяются в науке и технике. При помощи атомно-силового микроскопа исследуют не только проводящие, но и непроводящие поверхности. Из достоинств данной технологии можно выделить возможность получения высококачественных снимков микрорельефа и истинно трехмерного рельефа поверхности, а также возможность автоматизированного анализа результатов обработки. К ограничениям метода можно отнести возможное искажение изображения вследствие нелинейности сканера, относительно низкую скорость сканирования и небольшую область сканирования образцов / изделий [1].

При работе с АСМ изображениями, результаты сканирования подвергаются математической обработке, что позволяет определять различные параметры топографии поверхности. Эти параметры зависят определенным образом от материала и способа обработки поверхности. Принцип работы АСМ основан на прохождении электроном потенциального барьера, который образован разрывом электрической цепи (пространством между зондирующим микроострием и поверхностью образца). В основе работы прибора лежит феномен электронного туннелирования (туннельный эффект). В качестве зонда используется игла из нитрида кремния на упругой пластинке – кантилере, который деформируется под действием сил возникающих между иглой и поверхностью. Степень деформации кантилера определяется по отклонению луча лазера, отражаемого от тыльной зеркальной стороны кантилера [2].

В настоящее время микроструктура и свойства быстрозатвердевших (БЗ) сплавов Al-Cr активно исследуются, так как алюминиевые сплавы рассматриваются в качестве перспективных материалов для изготовления емкостей, предназначенных для хранения и транспортировки сжатого водородного топлива [3]. Цель данного исследования — определить зависимости между разными параметрами шероховатости поверхности БЗ сплавов алюминия в зависимости от поверхности анализа и обработки.

В работе использовались сплавы Al-Cr с расчётным содержанием хрома 1.0 ат.%, 1.5 ат.%, 3.0 ат.%. Высокая скорость охлаждения порядка  $10^6$  К/с была достигнута выплёскиванием расплавленной капли сплава на внутреннюю поверхность вращающегося медного цилиндра [3]. Для изучения отбирались фольги толщиной 30-60 мкм и шириной 5-10 мм. Типичные АСМ изображения поверхностей фольг приведены на рис. 1.

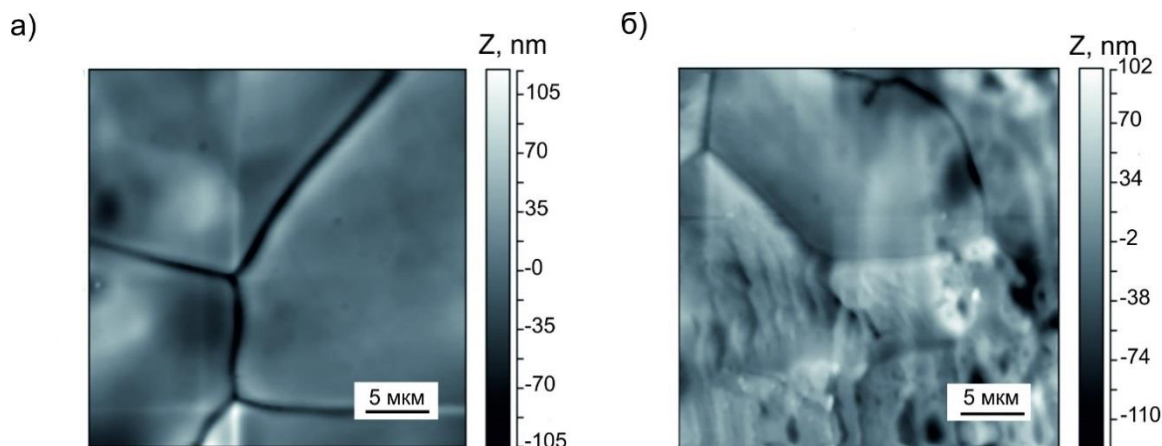


Рисунок 1 – Типичные изображения поверхности двух образцов сплава Al-1.5 ат. % Cr, полученные с помощью АСМ: а – свежезакаленная фольга, б – фольга, подвергнутая исследованию с помощью ТДС.

Методом АСМ с помощью атомного силового микроскопа NT-206 исследовалась топография обеих поверхностей фольг: поверхность А, которая контактировала с подложкой при затвердевании, и противоположная поверхность Б, которая контактировала с воздухом. Некоторые фольги подвергались анализу с помощью термодесорбционной спектроскопии (ТДС), когда образец отжигается с заданной скоростью нагрева от комнатной температуры до 600°C [3]. Для анализа на каждом АСМ изображении выбирали 11 сечений с шагом 2 мкм и для каждого из них определяли среднеарифметическую шероховатость  $R_a$ , высоту неровностей профиля  $R_z$ , среднее значение шага между выступами  $S$  и вычисляли коэффициент  $k = R_z/S$ , используемый для оценки неоднородности поверхности и несущий информацию о количественном соотношении поперечной и продольной шероховатости изучаемой поверхности [4].

На рис. 2 представлены значения параметров шероховатости, полученные после обработки АСМ изображений с помощью методов математической статистики. Установлено, что  $k$  и  $R_a$  зависят от содержания хрома и термической обработки сложным образом. Согласованное изменение параметров является в случае БЗ сплава, содержащего 3.0 ат. % Cr, что может быть объяснено достаточно высоким содержанием хрома, который, как известно, повышает термическую стабильность сплавов алюминия. Также интерес представляет обнаруженный факт, что с ростом содержания Cr коэффициент  $k$  уменьшается для поверхности А и увеличивается для поверхности Б. Для образцов, исследованных с помощью ТДС, наблюдается увеличение коэффициента  $k$  для поверхности А (за исключением сплава Al-1.0 ат. % Cr), а для поверхности Б коэффициент  $k$  увеличивается. Таким образом, результаты данной работы показывают перспективность использования АСМ изображений с целью получения дополнительной информации о поперечных и продольных характеристиках неоднородности рельефа поверхности в зависимости от условий получения материалов.

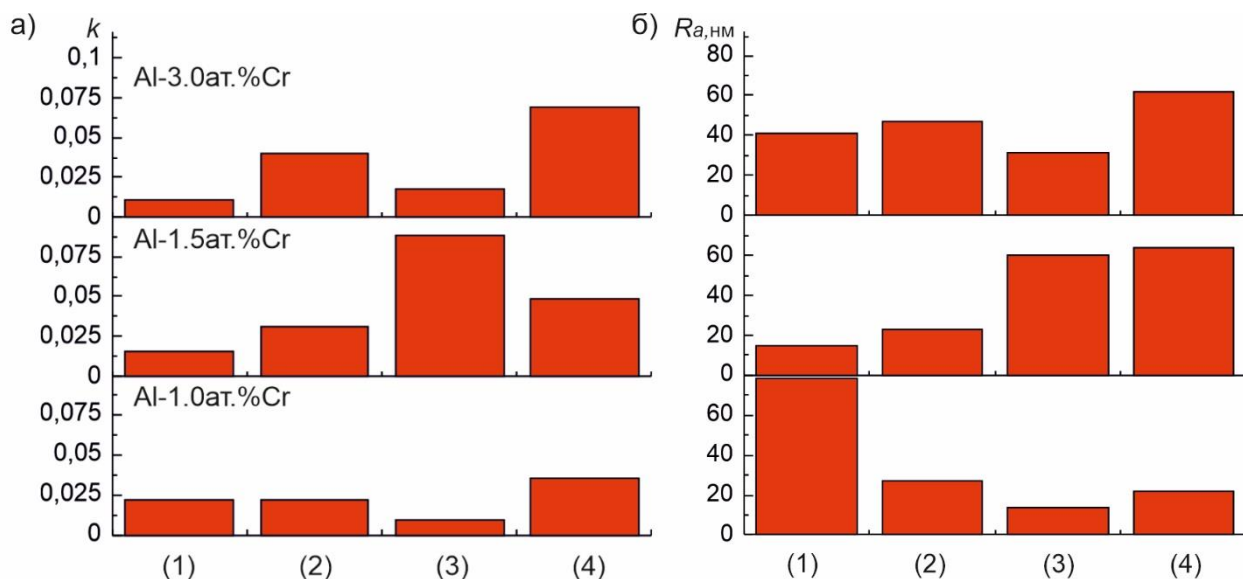


Рисунок 2 – Экспериментально измеренные величины  $k$  (а) и  $R_a$  (б) исследованных БЗ сплавов Al-Cr в зависимости от поверхности анализа и обработки: (1) и (3) - соответственно поверхности А и Б свежезакаленных образцов; (2) и (4) - соответственно поверхности А и Б фольг, исследованных с помощью ТДС.

**Список использованных источников:**

1. Rangelow, I. W. Atomic force microscope integrated with a scanning electron microscope for correlative nanofabrication and microscop / I. W. Rangelow, M. Kaestner, T. Ivanov, A. Ahmad // J. Vac. Sci. Technol. B. – 2018. – Vol. 36, № 6. – P. 33-45.
2. Саликова, С. П. Атомно-силовая микроскопия – новая возможность в изучении кардиомиоцитов / С. П. Саликова, А. А. Стадников, А. Н. Никиян // Вестник ОГУ. – 2003. - № 1. – С. 59-67.
3. Ташлыкова-Бушкевич, И. И. Анализ влияния структуры и состава на поведение водорода в быстрозатвердевших сплавах Al-Cr методом фотоэлектронной спектроскопии с использованием синхротронного излучения / И. И. Ташлыкова-Бушкевич // Актуальные проблемы физики твердого тела: материалы VII Междунар. науч. конф., Минск, 22 - 25 ноября 2016 г / НПЦ НАН Беларуси по материаловедению. – Минск : Ковчег, 2016. – С. 92-94.
4. Ташлыкова-Бушкевич, И. И. Количественный анализ нанорельефа поверхности металлических пленок на стекле по данным сканирующей зондовой микроскопии / И. И. Ташлыкова-Бушкевич, Е. С. Мойсейчик, Р. Д. Лобач, Д. В. Суходольский // Материалы и структуры современной электроники: материалы VIII Междунар. науч. конф., Минск, 10 - 11 октября 2018 г. / редкол.: В.Б. Оджаев (отв. ред.) [и др.]. – Минск: БГУ, 2018. – С. 111 – 117.

## АНАЛИЗ РЕЗУЛЬТАТОВ ПРИМЕНЕНИЯ ЭВРИСТИЧЕСКИХ ТЕХНОЛОГИЙ В ИЗУЧЕНИИ ФИЗИКИ НА ПОТОКЕ ВМСИС В БГУИР

Мельник Н.С., Филлипович В.М., Майнич Н.С., Патрушев А.С., Герус А.Е.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Ташлыкова-Бушкевич И.И. – к.ф.-м.н., доцент

В данной работе выполнен анализ эффективности применения эвристических технологий в БГУИР на факультете компьютерных систем и сетей (ФКСИС) в рамках лекционного курса общей физики в 2018/19 и 2019/20 учебных годах на примере потока специальности «Вычислительные машины, системы и сети» (ВМСИС) 850501-6. Полученные результаты наглядно показывают повышение уровня успеваемости студентов по физике в результате их привлечения к выполнению творческих проектов по физике.

На данный момент эвристическое обучение широко используется в образовательном процессе в школах. Под эвристическим обучением понимают способ обучения, при котором учащийся сам определяет смысл, цели и содержание образования [1]. Оно отличается от проблемного, т.к. цель проблемного состоит в усвоении учениками заданного предметного материала путём выдвижения учителем специальных познавательных задач-проблем [2]. Методика проблемного обучения построена так, что ученики «наводятся» учителем на известное решение или направление решения задачи. Эвристический же подход к образованию позволяет расширить возможности проблемного обучения, поскольку ориентирует преподавателя и учащегося на достижение неизвестного им заранее результата [1]. Эвристическое обучение — это обучение, «открывающее ученика»: «открытию «необходима» не выхолощенная и отобранная «правильная» информация, а реальность, окружающая учащегося» [3, с. 10]. При этом в ходе учебной деятельности ученик, познавая объекты действительности, создаёт свой образовательный продукт.

В БГУИР в качестве эксперимента элементы эвристического обучения применяются в лекционных занятиях по физике на потоке ВМСИС факультета КСИС доцентом И.И. Ташлыковой-Бушкевич [4]. Суть авторской педагогической технологии заключается в подготовке творческих проектов по изучаемому предмету в течение семестра в рамках самостоятельной управляемой работы студентов (см. рис 1), причём проект может быть как практическим (собранный схема из электронных компонентов либо установка), так и теоретическим. Как показывает практика, в течение учебного семестра от тридцати до сорока процентов студентов решают создавать собственные проекты. Тизеры этих проектов размещаются на YouTube канале «Эвристика в физике» [5].

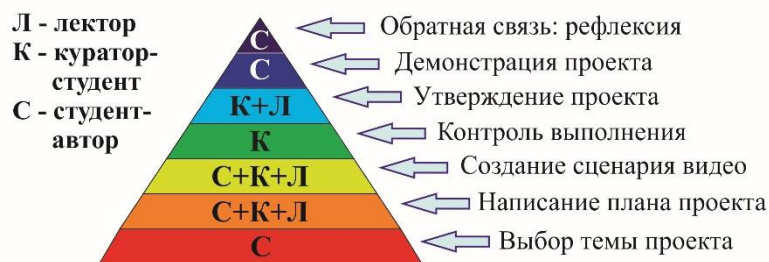


Рисунок 1 — Диаграмма, иллюстрирующая этапы создания творческих проектов по физике в БГУИР [3]

Целью применения эвристических методов обучения на потоке ВМСИС является привлечение студентов к углубленному изучению физики, а также повышение их успеваемости по курсу физики. Для оценки эффективности использования разработанной Ташлыковой-Бушкевич И.И. технологии с элементами эвристического обучения в данном исследовании представлено сравнение результатов трех экзаменационных сессий потока 850501-6 по физике. При этом рассмотрена успеваемость студентов за первый семестр (период до начала участия студентов в творческих проектах), а также за второй и третий семестры изучения физики (период выполнения творческих проектов).

Из результатов экзамена по физике I семестра (зимняя сессия 2018/2019 уч. г.) на рис. 2а следует, что около 30% студентов сдали экзамен на отметку «4», в то время как процент студентов, получивших отметки «6», «7», «8», «9» и «10», колеблется в диапазоне от 6% до 12%. На рис. 2б представлено распределение баллов экзамена по физике данного потока по результатам II семестра (летняя сессия 2018/2019 уч. г.). Рисунок 2в показывает распределение баллов за экзамен в III семестре (зимняя сессия 2019/2020 уч. г.). Как мы видим, количество отметок «8», «9» и «10» значительно выросло при уменьшении суммарной доли отметок «4» и «5» в 6 раз по сравнению с I семестром. Также наблюдается рост числа отметок «6» и «7», что указывает на увеличение среднего уровня знаний по физике.

Анализ выполненных проектов показывает, что в подавляющем большинстве случаев студенты предпочитают практические проекты по сравнению с теоретическими работами, что связано с их желанием увидеть практическую реализацию изученного материала. Создание проектов по физике позволяет студентам не только изучить теорию различных изучаемых явлений, тем самым создавая необходимый фундамент для

написания проекта, а также развивает у них такие важные для будущей профессиональной деятельности навыки, как работа в команде, владение программным обеспечением для обработки и редактирования аудио и видео, работа с радиоэлектроникой и многое другое. Студенты представляют свои творческие проекты на лекциях или онлайн. В дальнейшем их работы используются в качестве обучающих материалов.

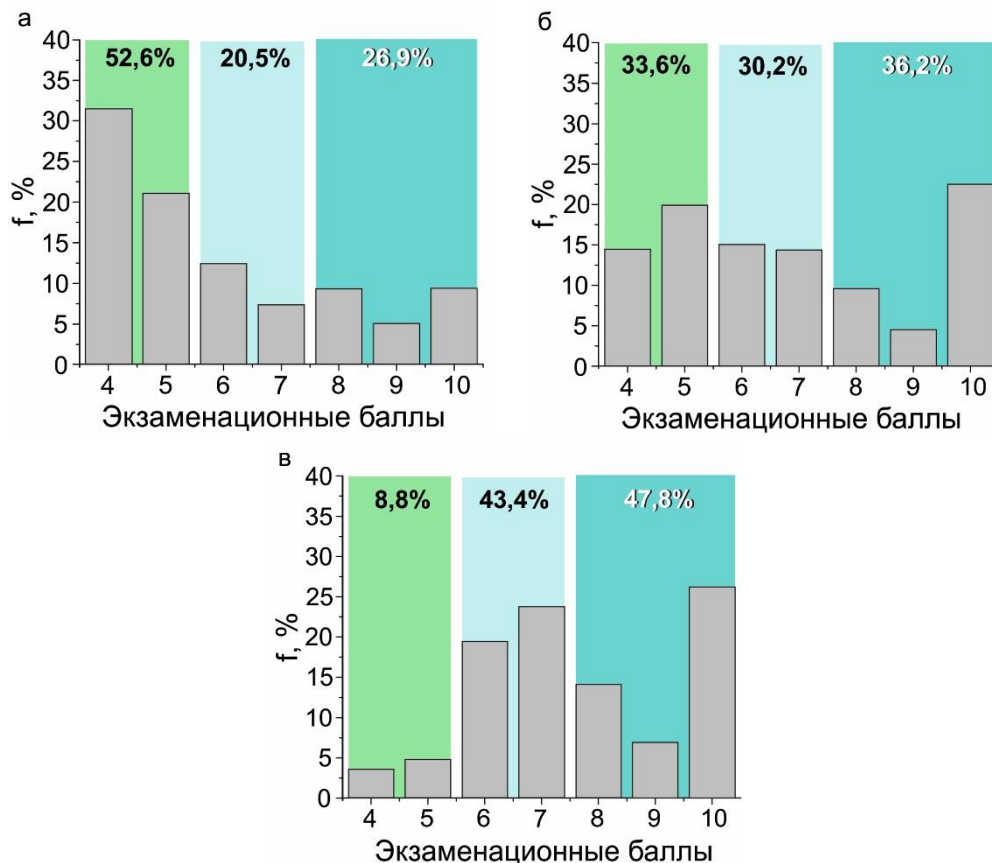


Рисунок 2 — Экзаменационные отметки в баллах по физике потока 850501-6 (без учета пересдач) за зимнюю сессию 2018/2019 уч.г. (а), летнюю сессию 2018/2019 уч.г (б) и зимнюю сессию 2019/2020 уч.г. (в)

Согласно полученным данным, мы можем сделать вывод о положительном влиянии внедрения эвристических технологий в процесс обучения на итоговую успеваемость студентов по физике. Процесс создания проектов привлекает студентов к научной и творческой деятельности, повышает их мотивацию изучать физику и формирует практические профессиональные компетенции студентов в рамках трехсеместрового курса общей физики.

**Список использованных источников:**

1. Научная школа А. В. Хуторского. Эвристическое обучение [Электронный ресурс]. — Режим доступа: <https://cutt.ly/5tQ68F9> — Дата доступа: 01.03.2020.
2. Абушкин, Х. Х. Методика проблемного обучения физике: учебное пособие для среднего профессионального образования / Х. Х. Абушкин. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2019. — 178 с.
3. Король, А. Д. Молчание в обучении: методологические и дидактические основы / А. Д. Король // Педагогика. — 2019. — № 4. — С. 5–14.
4. Ташлыкова-Бушкевич, И. И. Апробация авторской технологии организации лекционных занятий со студентами по физике с элементами эвристического обучения / И. И. Ташлыкова-Бушкевич // Вышэйшая школа. — 2019. — № 2. — С. 43–48.
5. YouTube-канал “Эвристика в физике” [Электронный ресурс] — <http://tiny.cc/0jbnaz> - Дата доступа: 10.03.2020.

## ПРИМЕНЕНИЕ РЕНТГЕНОСТРУКТУРНОГО АНАЛИЗА ДЛЯ ИЗУЧЕНИЯ ГИДРОКСИДОВ В ФОЛЬГАХ СПЛАВОВ Al-Cr

Романчук А.В., Адинцов Н.В., Япс А.Р., Кочерга М.В., Рагиня В.Ю.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Ташлыкова-Бушкевич И.И. – канд. физ.-мат. наук, доцент

В данной работе рассмотрено применение метода рентгеноструктурного анализа для изучения формирования гидроксидов в фольгах сплавов Al-Cr, полученных сверхбыстрой закалкой из расплава. Определено присутствие фаз, соответствующих гиббситу, диаспору и бемиту. Выявлено, что, начиная с температуры отжига 200°C до 300°C гиббсит, распадаясь, частично превращается в бемит. Вследствие распада гиббсита после 300°C обнаружены следы фазы  $\alpha - Al_2O_3$ . Результаты доказывают эффективность исследования фазовых превращений в сплавах алюминия в процессе термической обработки методом рентгеновской дифракции.

В настоящее время в промышленности широко используются методы, обеспечивающие получение материалов при неравновесных условиях. Одним из них является сверхбыстрая закалка из расплава (СБЗР) [1]. В результате СБЗР получают метастабильные материалы, в которых возможно формирование фаз, отсутствующих в образцах, полученных при равновесных условиях. Поэтому рентгеноструктурный анализ быстрозатвердевшего (БЗ) сплава начинается с поиска информации о том, какие фазы могут в нём формироваться при кристаллизации. Для их идентификации используются международные базы данных, такие как ICDD, ICSD и др. Однако, если фаза метастабильна, то её химический состав может быть переменным и не исключено, что информация о ней в базах отсутствует. Международный центр дифракционных данных (ICDD) – это научная организация, создавшая банк данных порошковой дифракции (PDF), который используется для идентификации кристаллических материалов. На конец 2019 года он содержал более 893 400 уникальных наборов данных, каждый из которых содержит дифракционную, кристаллографическую и библиографическую информацию, а также экспериментальные, приборные, выборочные условия и отдельные физические свойства разных фаз [2]. База данных неорганических кристаллических структур (ICSD) является крупнейшим в мире источником информации о кристаллическом строении неорганических веществ. Информация ICSD постоянно актуализируется: каждый год базу пополняют около 6000 новых структур [3].

В основе рентгеноструктурного анализа лежит явление дифракции рентгеновских лучей на трёхмерной кристаллической решётке. В качестве образца используется или порошок, нанесённый на поверхность специальной кюветы, или срез массивного поликристалла. Режим съёмки дифрактометра устанавливается специальной программой. Необходимо подобрать такие условия, которые позволили бы при достаточно большой интенсивности рефлексов получить хорошую точность в определении их углового положения. Полученная дифрактограмма содержит информацию о положении и интенсивности дифракционных максимумов. Положение пика задаётся углом отражения  $2\theta$ , а интенсивность – высотой сигнала [4].

При изучении материалов, полученных при неравновесных условиях, важно рассматривать образцы, прошедшие различную термическую обработку, поскольку с повышением температуры метастабильные фазы распадаются на более стабильные, что приводит к изменению фазового состава. Исследование БЗ сплавов при различных температурах позволяет оценить динамику процессов фазовых превращений.

В данной работе представлены экспериментальные результаты применения рентгеноструктурного анализа для изучения гидроксидов в фольгах сплавов Al-Cr. Как известно, материалы из сплавов алюминия активно используются в водородной энергетике [5] для хранения и транспортировки газообразного водорода под давлением, как альтернативного топлива. Высокопрочные сплавы алюминия при повышенных давлениях подвержены водородному охрупчиванию, поэтому в настоящее время экспериментальные и теоретические исследования этих сплавов проводятся для решения данной проблемы водородной энергетики [6,7].

Для получения сплавов Al-Cr, содержащих от 1.0 ат. % Cr до 3.0 ат. % Cr, был применен метод центробежной закалки расплава при частоте вращения медного цилиндра 1500 об/мин [8]. Скорость охлаждения расплава составила не меньше  $10^6$  К/с. Толщина полученных фольг была 50-100 мкм, ширина – 5-10 мм. Рентгеноструктурный анализ был проведён для свежезакалённых образцов и фольг, отожжённых в интервале температур от 20°C до 500°C, на рентгеновском дифрактометре Rigaku Ultima IV (U=40 кВ, I=40 мА) с использованием Cu K $\alpha$ -излучения ( $\lambda=0.15406$  нм). Для идентификации фаз была использована база данных ICDD PDF-2 (2013).

На рис. 1 показаны изменения фазового состава сплава Al-3.0 ат. % Cr в процессе термической обработки. На дифрактограммах кроме линий интенсивности гидроксидов представлены также линии фазы  $\alpha - Al_2O_3$  и продукты распада гидроксидов при отжиге. Было установлено, что в сплаве Al-3.0 ат. % Cr, полученном СБЗР, присутствуют фазы, соответствующие гиббситу ( $\gamma - Al(OH)_3$ ),



диаспору ( $\alpha - \text{AlOOH}$ ) и бемиту ( $\gamma - \text{AlOOH}$ ). Следует отметить, что в интервале температур от  $200^\circ\text{C}$  до  $300^\circ\text{C}$  интенсивность дифракционных отражений бемита возрастает, а гиббсита значительно снижается. Полученные данные свидетельствуют о том, что при температуре отжига выше  $200^\circ\text{C}$  гиббсит, распадаясь, частично превращается в бемит. Как результат распада гиббсита после  $300^\circ\text{C}$  в образцах обнаружено присутствие фазы  $\alpha - \text{Al}_2\text{O}_3$ . Отметим, что результаты рентгеноструктурного анализа дополняют данные, полученные ранее методом сканирующей фотоэлектронной спектроскопии, используя синхротронное излучение [8].

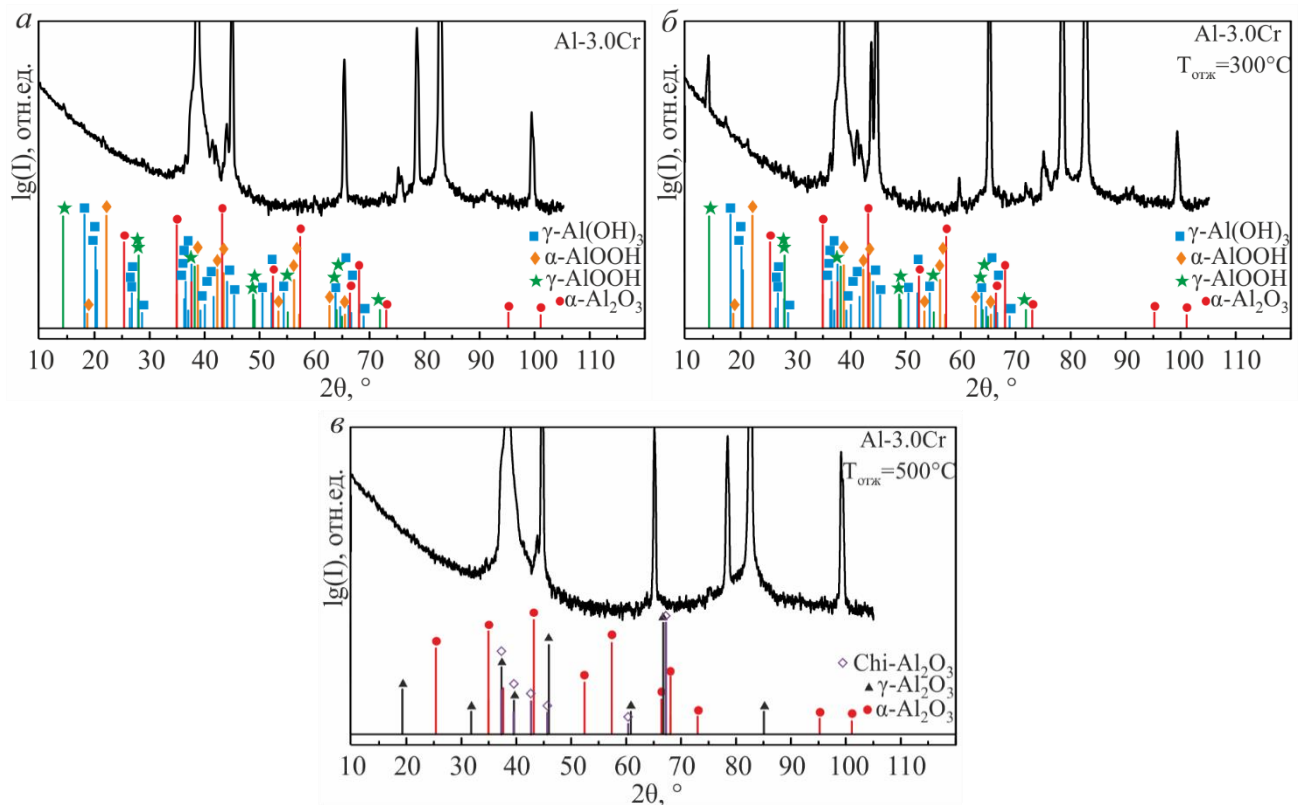


Рисунок 1 – Типичные дифрактограммы БЗ сплава Al-3.0 ат. % Cr: а – свежезакалённая фольга, б и в – фольги, отожжённые соответственно при  $300^\circ\text{C}$  и  $500^\circ\text{C}$ . На штрихдиаграммах показаны проиндексированные линии с интенсивностью выше 10 усл. ед.

Таким образом, полученные в работе результаты демонстрируют эффективность применения метода рентгеновской дифракции для определения с достаточно высокой точностью фазовых переходов в БЗ сплавах алюминия с целью изучения вопросов их термической стабильности. Подход, описанный в данной работе, позволит в дальнейших исследованиях лучше понять и определить возможности применения сверхбыстрой закалки для улучшения свойств высокопрочных сплавов алюминия при их практическом использовании в области водородной энергетики.

**Список использованных источников:**

1. Lavernia, E. J. Rapid solidification processing with specific application to aluminium alloys / E. J. Lavernia, J. D. Ayers, T. S. Srivatsan // *International Materials Reviews*. – 1992. – Vol. 37, № 1. – P. 1–44.
2. Международный центр дифракционных данных [Электронный ресурс]. – Режим доступа: <http://www.icdd.com>. – Дата доступа: 30.03.2020.
3. База данных неорганических кристаллических структур [Электронный ресурс]. – Режим доступа: <http://www.fiz-karlsruhe.de/icsd>. – Дата доступа: 30.03.2020.
4. Горелик, С. С. Рентгенографический и электронно-оптический анализ / С. С. Горелик, Л. Н. Расторгуев, Ю. А. Скаков. – М.: Металлургия, 1970. – 368 с.
5. Staffell, I. The role of hydrogen and fuel cells in the global energy system / I. Staffell, D. Scamman, A. V. Abad, P. Balcombe, P. E. Dodds, P. Ekins, N. Shan, K. R. Ward // *Energy Environ. Sci.* – 2019. – Vol. 11, № 2. – P. 463–491.
6. Yamabe, J. Surface coating with a high resistance to hydrogen entry under high-pressure hydrogen-gas environment / J. Yamabe, S. Matsuoka, Y. Murakami // *International Journal of Hydrogen Energy*. – 2013. – Vol. 38, № 24. – P. 10141–10154.
7. Scully, J. R. Hydrogen embrittlement of aluminum and aluminum-based alloys / J. R. Scully, G. A. Young, S. W. Smith // *Gaseous Hydrogen Embrittlement of Materials in Energy Technologies*. Vol. 1. / ed. R. P. Gangloff, B. P. Somerday. – Cambridge, 2012. – Ch. 19 – P. 707–768.
8. Tashlykova-Bushkevich, I. I. Hydrogen behavior in Al-Cr alloys: synchrotron-based photoelectron microscopy of the rapidly solidified structure / I. I. Tashlykova-Bushkevich, M. Amati, B. Alemán, H. Sezen, L. Gregoratti, M. Kiskinova // *International Journal of Hydrogen Energy*. – 2016. – Vol. 41, № 21. – P. 9100–9107.

## МОДЕЛИРОВАНИЕ ЭФФЕКТА ХОЛЛА В ПОЛУПРОВОДНИКАХ С ПОМОЩЬЮ СРЕДЫ MATLAB

Рафеев П.Ю.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Григорьев А. А. - доцент, канд. физ.-мат. наук

В современных научных исследованиях всё большую роль, помимо традиционных наблюдения и эксперимента, начинает играть симуляция. Компьютерные программы позволяют качественно наблюдать явление без необходимости в финансовых расходах на оборудование, реактивы и всё остальное, необходимое для проведения эксперимента. Поэтому актуальным становится поиск и разработка математических моделей, описывающих природные явления с различной степенью точности, что и является целью данной работы: построить модель эффекта Холла, реализовать её в среде MATLAB и сравнить с экспериментальными данными для проверки её точности.

Эффект Холла — гальваномагнитное явление возникновения поперечной разности потенциалов при помещении проводника с постоянным током в магнитное поле. Причиной его является смещение движущихся заряженных частиц к границам проводника под действием силы Лоренца. Если же в магнитное поле поместить полупроводник с постоянным током, то по знаку разности потенциалов между поперечными его краями можно судить о типе его проводимости, так как и электроны, и дырки смещаются в одну сторону.

Моделировать данное явление можно на примере германиевой пластины с примесью мышьяка шириной  $b$  и толщиной  $a$  и текущим через неё током  $I$ , плотность которого  $j$ . Магнитное поле  $B$ , перпендикулярное плоскости пластины, будет создавать поперечный ток в проводнике и разность потенциалов между краями этой пластины (рисунок 1).

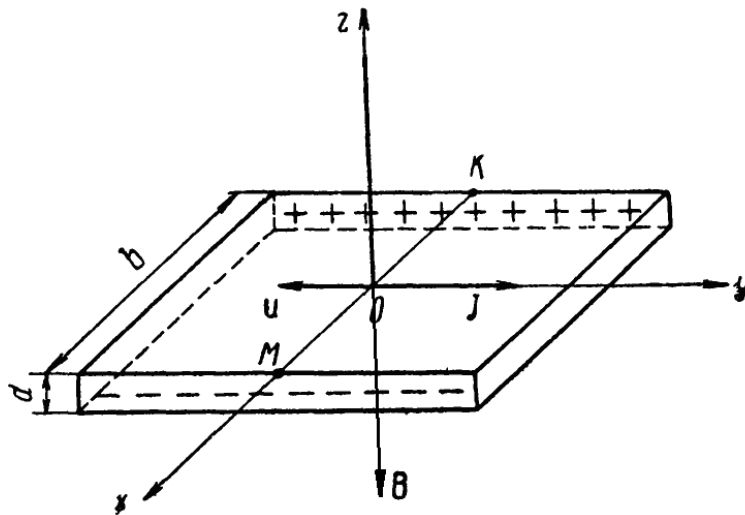


Рисунок 1 – Схема модельной установки и расположение векторов в экспериментальной схеме

После составления уравнений движения заряженных частиц, для их численного решения лучше воспользоваться программой MATLAB — пакетом прикладных программ для решения задач технических вычислений и моделирования. Конечным расчётным параметром следует взять величину поперечного тока в начальный момент времени, так как его будет легко измерить экспериментально (соединив края пластины через амперметр).

Таким образом, выполнив моделирование и проведя эксперимент, получим следующую таблицу значений поперечного тока в зависимости от продольного тока (при  $B=1.0$  Тл и  $\sigma = 800$   $1/\text{Ом}\cdot\text{м}$  для образца с примесью мышьяка):

Таблица 1. Численные значения токов, полученные в результате моделирования и эксперимента.

Модельные значения		Экспериментальные значения	
Тока в проводнике, мА	Поперечный ток, мА	Ток в проводнике, мА	Поперечный ток, мА
0	0	0	0
0.2	-0.04	0.2	-0.04
0.4	-0.08	0.4	-0.09
0.5	-0.12	0.5	-0.13
0.7	-0.16	0.7	-0.16
0.9	-0.20	0.9	-0.20
1.1	-0.24	1.1	-0.25
1.2	-0.29	1.2	-0.31
1.4	-0.33	1.4	-0.34
1.6	-0.37	1.6	-0.38
1.8	-0.41	1.8	-0.43

В итоге наблюдается хорошее совпадение результатов моделирования и эксперимента, что говорит о качественно построенной и математически верно описанной модели. Тип проводимости - электронный (на что явно указывает знак тока смещения), как и следовало ожидать у образца с небольшой примесью мышьяка.

Построение и анализ графиков зависимостей поперечного тока от продольного в среде MATLAB также позволяет заметить, что в целом, поперечный ток прямопропорционален току в проводнике. Это обусловлено тем, что при увеличении подаваемого напряжения, скорость движения электронов и дырок увеличивается, из-за чего воздействие магнитного поля на них также увеличивается, что вызывает и увеличение разности потенциалов между стенками проводника, для компенсации всех воздействий. Постоянное же превышение экспериментально измеренным поперечным током соответствующего модельного значения обусловлено появлением вблизи проводника его собственного магнитного поля, которое также создаёт поперечную разность потенциалов (хоть и вклад его на два порядка меньше, чем вклад внешнего магнитного поля).

**Список использованных источников:**

1. Кучис Е. В. Гальваномагнитные эффекты и методы их исследования — М.: Радио и связь, 1990 — 615 с.
2. Дьяконов В. П. Справочник по применению системы PC MATLAB. — М.: «Физматлит», 1993. — 112 с.

**СЕКЦИЯ «ВСТРАИВАЕМЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ  
РЕАЛЬНОГО ВРЕМЕНИ»**

# ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ РАЗРЕЖЕННОГО БПФ ДЛЯ ОБРАБОТКИ ЗВУКОВЫХ СИГНАЛОВ НА МОБИЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ ПЛАТФОРМАХ

Бессмертный Н.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Лихачев Д.С. – к.т.н., доцент

Преобразование Фурье является базовым алгоритмом обработки данных и имеет применения во многих инженерных областях. Алгоритм Кули и Тьюки для вычисления разреженного БПФ считается одним из десяти важнейших алгоритмов двадцатого столетия — его открытие в 1965 г. привело к революции в алгоритмах обработки сигналов, и этот алгоритм применяется повсеместно на сегодняшний день.

Дискретное преобразование Фурье является одной из наиболее важных и широко используемых вычислительных задач. Его область применения широка и включает в себя обработку сигналов, коммуникацию и сжатие аудио/изображений/видео. Следовательно, быстрые алгоритмы для DFT очень ценны. В настоящее время самым быстрым таким алгоритмом является быстрое преобразование Фурье (FFT), которое вычисляет DFT  $n$ -мерного сигнала за  $O(n \log n)$  время.

Алгоритм БПФ используется со случайными сигналами, однако сигналы, встречающиеся в реальной жизни, зачастую обладают некоторыми свойствами, позволяющими ускорить вычисление. Одно из этих свойств – это разреженность спектра: спектр Фурье сигналов, встречающихся в реальной жизни, часто хорошо приближается малым количеством коэффициентов, это свойство лежит в основе многих схем сжатия звука, изображений и видео, например JPEG. Рассмотрим сжатие JPEG, мы представляем картинку как двумерный сигнал, вычисляем БПФ сигнала и так как мы хотим достичь сжатия, мы сохраняем малое количество  $k$  доминирующих компонент Фурье. Сложность такого сжатия будет  $O(N \cdot \log N)$  при применении БПФ и потом за линейное время  $O(N)$  мы можем найти доминантные  $k$  компоненты. Если применить алгоритм разреженного БПФ то он будет вычислять  $k$  доминантные компоненты напрямую и сложность будет  $O(k \cdot \log N)$ . Более того, в данном случае время работы алгоритма сублинейно по  $N$ , т.е. алгоритм даже не считает все  $N$  элементов сигнала во временной области.

Рассмотрим сигнал  $x$  размера  $n$ , ДПФ которого  $\hat{x}$  определено как:

$$\hat{x} = \sum_{t=0}^{n-1} x(t) * e^{-j2\pi ft/n} \quad (1),$$

$\hat{x}$  считается  $k$ -разреженным, если он имеет  $k$  отличных от нуля частотных коэффициентов, а оставшиеся  $n$  коэффициентов равны нулю. Задача разреженного БПФ: точное восстановление  $\hat{x}$  с целью нахождения частотных положений  $f$  и значений  $\hat{x}(f)$  для  $k$  ненулевых коэффициентов. Для общих сигналов разреженное БПФ вычисляют как  $k$ -разреженное приближение  $\hat{x}^k$  от  $\hat{x}$ . Наилучшее  $k$ -разреженное приближение можно получить, определив все, кроме наибольших  $k$ , коэффициентов  $\hat{x}$  равными 0. Целью является расчет приближения  $\hat{x}^k$ , в котором ошибка приближения ограничена ошибкой на лучшем  $k$ -разреженном приближении. Формально должно удовлетворять следующим условиям:

$$\|\hat{x} - \hat{x}^k\|_2 \leq C \min_{k\text{-sparse } y} \|\hat{x} - y\|_2 \quad (2),$$

где  $C$  - некоторый фактор приближения, минимизация по точности  $k$ -разреженных сигналов.

Применяя алгоритм разреженного БПФ были получены следующие тестовые результаты в сравнении с математической моделью БПФ:

- при размере сигнала  $N=65536$  и разреженности  $k=1$ , время выполнения разреженного БПФ составило 0.001440 секунды, время выполнения БПФ – 0.004964 секунды.
- при размере сигнала  $N=65536$  и разреженности  $k=50$ , время выполнения разреженного БПФ составило 0.002796 секунды, время выполнения БПФ - 0.004118 секунды.
- при размере сигнала  $N=1048576$  и разреженности  $k=50$ , время выполнения разреженного БПФ составило 0.012571 секунды, время выполнения БПФ – 0.169344 секунды.

Можно сделать вывод, что чем больше  $N$  и чем  $N > k$ , тем быстрее алгоритм разреженного БПФ в сравнении с БПФ.

#### Список использованных источников:

1. FFTW[Электронный ресурс]. – Режим доступа: <http://www.fftw.org/>. – Дата доступа: 09.03.2020.
2. SFFT: Sparse Fast Fourier Transform [Электронный ресурс]. – Режим доступа: <http://groups.csail.mit.edu/netmit/sFFT/>. – Дата доступа: 09.03.2020.
3. Computer Science Center[Электронный ресурс]. – Режим доступа: <https://compscicenter.ru/courses/sparsefft/2015-autumn/> Дата доступа: 10.03.2020.

# РЕАЛИЗАЦИЯ ХЭШ ФУНКЦИЙ И СПОСОБЫ ИХ ВЫЧИСЛЕНИЯ

Гридюшко А.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Качиснский М.В. – к.т.н., доцент

Функцией хэширования называется отображение  $H:V^* \rightarrow V_n$ , где  $n \in \mathbb{N}$  — натуральное число,  $V^*$  — множество всех двоичных векторов (строк) конечной размерности (включая пустую строку),  $V_n$  — множество всех  $n$ -мерных двоичных векторов. Аргумент  $M \in V^*$  функции хэширования называется сообщением. Частью сообщения называется любой подвектор сообщения  $M$ . Хэш-кодом (хэш-значением) сообщения  $M \in V^*$  называется значение  $H(M) \in V_n$ .

На сегодняшний день наиболее распространены функции хэширования, построенные по итеративному принципу (итеративные функции хэширования). Он восходит к работам [6, 7] Р. Меркля (Ralph C. Merkle) и И. Дамгорда (Ivan Damgard) 1989 года, в связи с чем итеративная конструкция получила название конструкции Меркля–Дамгорда. Конструкция Меркля – Дамгорда позволяет свести задачу построения хэш-функции  $H:V^* \rightarrow V_n$  к задаче построения отображения  $g:V_q \times V_1 \rightarrow V_q$  с определенными свойствами. По итеративному принципу построены такие хэш-функции, как MD5 [8], SHA-1 [3, 4], семейство хэш-функций SHA-2 [4], хэш-функции ГОСТ Р 34.11-94 и ГОСТ Р 34.11-2012. Опишем итеративные хэш-функции в общем виде и введем обозначения, которые будут использоваться в дальнейшем. Перед вычислением значения итеративной функции хэширования исходное сообщение  $M \in V^*$  преобразуется в сообщение  $M \in V^*$ , длина которого кратна  $l$ . Данное преобразование называется процедурой дополнения сообщения (message padding), а сообщение  $M$  — расширенным сообщением (padded message). Дополнение сообщения может осуществляться различными способами, но, как правило, включает два этапа: конкатенация сообщения со значением его длины, записанным в двоичном виде, и с некоторой последовательностью до длины, кратной  $l$ , так, чтобы каждому исходному сообщению соответствовало единственное расширенное сообщение. Пусть теперь  $M$  — сообщение, полученное с помощью процедуры дополнения сообщения  $M$ , и  $M = m_1 \| m_2 \| \dots \| m_t$ ,  $m_i \in V_l$ ,  $i = 1, \dots, t$ . Векторы  $m_i \in V_l$ ,  $i = 1, \dots, t$ , называются блоками сообщения  $M$ . Алгоритм вычисления хэш-кода  $h = H(M)$  итеративной функции хэширования состоит из последовательного вычисления значений так называемой функции сжатия (compression function)  $g:V_q \times V_1 \rightarrow V_q$  от результата предыдущей итерации и очередного блока сообщения:

$$h_i = g(h_{i-1}, m_i), \quad i = 1, \dots, t, \quad (1)$$

где  $h_0$  равно заданному фиксированному значению из множества  $V_q$ . Значение  $h_0$  принято называть инициализационным вектором (initialization vector, initial value, IV). Значение хэш-кода  $h$  полагается равным  $f(h_t, M)$ , где  $f:V_q \times V^* \rightarrow V_n$  — некоторое заключительное преобразование. Завершающее преобразование  $f$  часто не зависит от  $M$  и является тождественным преобразованием первого аргумента (в этом случае  $q=n$ ) или операцией редуцирования первого аргумента (в этом случае  $q>n$ ). Для хэш-функции ГОСТ Р 34.11-94 завершающее преобразование заключается в применении функции сжатия к блоку, являющемуся суммой по модулю  $2^{256}$  всех блоков сообщения:  $f(h_t, M) = g(h_t, \Sigma)$ ,  $\Sigma = m_1 \boxplus \dots \boxplus m_t$ . Данный подход эквивалентен дополнению исходного сообщения помимо двоичной записи длины значением  $\Sigma$ , которое принято называть контрольной суммой блоков сообщения.

#### Список использованных источников:

1. Колчин В.Ф., Севастьянов Б.А., Чистяков В.П. Случайные размещения. — М.: Наука, 1976.
2. Матюхин Д.В., Шишкин В.А. О криптографической стойкости хэш-функции ГОСТ Р 24.11-94 // Обзор. прикл. и промыш. матем. — 2010. — Т. 17. — С. 750–751.
3. FIPS PUB 180-1. Secure hash standard. — April, 1995. — <http://www.itl.nist.gov/fipspubs/fip180-1.htm>.
4. FIPS PUB 180-2. Secure hash standard. — August, 2002. — <http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf>
5. Aumasson J.-P. Faster multicollisions // INDOCRYPT'08. Lect. Notes Comput. Sci. — 2008. — V. 5365. — P. 67–77.
6. Damgaard I. A design principle for hash functions // CRYPTO'89. Lect. Notes Comput. Sci. — 1990. — V. 435. — P. 416–427.
7. Merkle R.C. One way hash functions and DES // CRYPTO'89. Lect. Notes Comput. Sci. — 1990. — V. 435. — P. 428–446.
8. Rivest R. The MD5 message-digest algorithm. Request for comments (RFC)1321, Internet Activities Board, Internet Privacy Task Force, April 1992.

# АДАПТИВНАЯ ФИЛЬТРАЦИЯ ПРИ ОБРАБОТКЕ СИГНАЛОВ В ТРАКТАХ РЛС

Егоров А.В., Конопелько Я.Д.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Матюшков А.Л. – к.т.н., доцент

В работе рассматриваются основные принципы функционирования адаптивных антенных решеток. Изучаются алгоритмы пространственной фильтрации сигналов в приемных трактах РЛС с адаптивной антенной решеткой. Расчет весовых коэффициентов осуществляется на основе рекурсивного алгоритма наименьших квадратов (RMS). В работе решается задача математического моделирования с целью подтверждения эффективности рассмотренного метода адаптивной фильтрации шумовых помех.

Антенная решетка (АР) – в общем виде представляет собой набор антенных элементов определенным образом размещенных в пространстве.

Адаптивной антенная решетка становится благодаря алгоритмам пространственной обработки сигнала (рисунок 1). Эти алгоритмы анализируют принимаемый сигнал и рассчитывают весовые коэффициенты, значение которых позволяют менять форму диаграммы направленности АР.

Адаптивная пространственная фильтрация позволяет эффективно решать задачи подавления помех в приемных трактах. Это достигается путем адаптивного формирования диаграммы направленности АР с провалами в направлениях на источники помех и поддержания требуемого уровня отношения сигнал/шум.

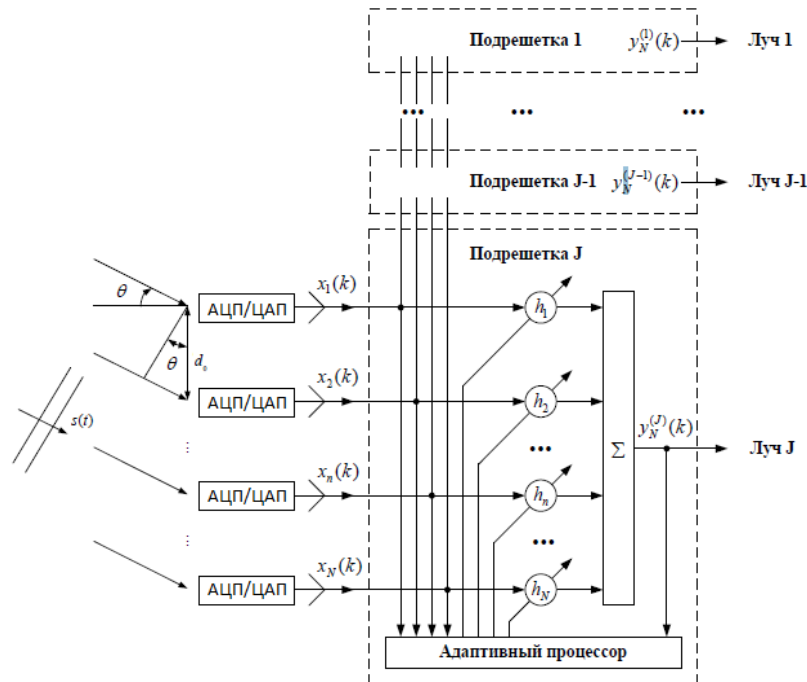


Рисунок 1 – Структура адаптивной антенной решетки

В адаптивных АР для упрощения процесса формирования опорного сигнала для фильтров применяется входной сигнал с константным значением модуля информационных символов (СМ-сигнал). [4]

В общем случае адаптивная обработка СМ-сигналов основывается на минимизации многоэкстремальной целевой функции:

$$J(p, q) = E[|s^p - |y(k)|^p|^q], \quad (1)$$

где  $s = |a_i| = \sqrt{a_i^* a_i} = const$  – значение модуля информационных символов  $a_i$ . [3];

$y(k) = h_N^H(k)x_N(k)$  - выходной сигнал АР;

$k$  – номер итерации алгоритма адаптивной фильтрации;

$p$  и  $q$  – параметры, используемые в обозначении СМ-алгоритмов и их целевых функций;

Из работы [4] следует, что если  $p = q = 2$ , то: многоэкстремальную функцию (1) можно

преобразовать в квадратичную унимодальную функцию:

$$J'(2,2) = E \left[ |s^2 - h_N^H(k)z_N(k)|^2 \right], \quad (2)$$

$$\text{где } z_N(k) = x_N(k)x_N^H(k)h_N(k-1) = x_N(k)y^*(k), \quad (3)$$

Совместное применение целевой функции (2) и линейных ограничений (LC), широко используются в алгоритмах адаптивной обработки сигналов в цифровых АР, рассмотренных в этой работе.

Применяемые в данной работе адаптивные фильтры основаны на работе LC RLS (Recursive Least Square).

Вектор комплексных весовых коэффициентов  $h_N(k)$ , вычисляемый на базе таких алгоритмов имеет вид:

$$h_N(k) = \hat{R}_N^{-1}(k)r_N(k) + \hat{R}_N^{-1}(k)C_{NJ}[C_{NJ}^H\hat{R}_N^{-1}C_{NJ}]^{-1}[f_j - C_{NJ}^H\hat{R}_N^{-1}(k)\tilde{r}_N(k)] \quad (4)$$

Работоспособность и функциональная эффективность алгоритмов были подтверждены моделированием адаптивной АР с числом радиоэлементов  $N = 8$ , расстоянием между ними  $d_0 = 0.5 \lambda_0$ . 3 луча АР были ориентированы  $0^\circ, 20^\circ$  и  $40^\circ$  в направлении источников сигналов. Помеха, представленная в виде белого гауссова шума моделировалась с направления  $-40^\circ$

Результаты моделирования представлены на рисунках 2,3

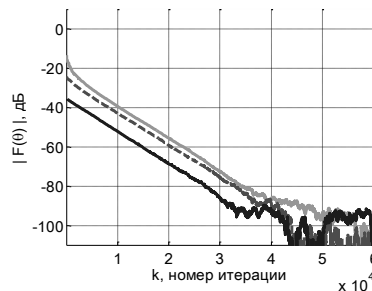


Рисунок 2 – Подавление уровня не коррелируемой помехи

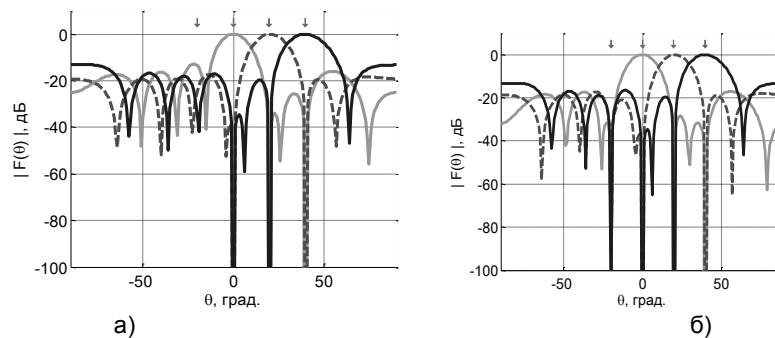


Рисунок 3 – Формирование провалов ДН на источник помехи: а) – в начальном состоянии; б) – после адаптивной фильтрации

Из рисунков мы можем увидеть, что применение алгоритмов адаптивной пространственной фильтрации в АР позволяет эффективно подавлять некоррелируемые помехи в приемном тракте, путем формирования провалов в ДН, при сохранении необходимого уровня основного лепестка.

**Список использованных источников:**

1. Григорьев, Л.Н. Цифровое формирование диаграммы направленности в фазированных антенных решетках/ Л.Н. Григорьев. – М.: Радиотехника, 2010 – 144 с.
2. Активные фазированные антенные решетки/ Под ред. Д.И. Воскресенского и А.И. Канащенкова. – М.: Радиотехника, 2004. – 408с.
3. Benvenuto N, Cherubini G. Algorithms for communication systems and their applications, Hoboken: John Wiley and Sons, Inc., 2002. – 1285 p.
4. Chen Y., Le-Ngoc T., Champagn B., Xu C. Recursive least squares constant modulus algorithm for blind adaptive array // IEEE Trans. Signal Processing. – 1985. – Vol. 33. – No. 2. – P. 1452–1456.
5. Джиган В.И. Вычислительно эффективный линейно-ограниченный комплексный RLS-алгоритм в арифметике действительных чисел // Доклады 14-й Международной конференции «Цифровая обработка сигналов и ее применения (DSPA-2012)» (Российская академия наук: Институт проблем управления им. В.А. Трапезникова, 28 – 30 марта 2012 г.). – Москва, 2012. – Том 1. – С. 77–81.



## БАЗОВЫЕ ОСНОВЫ ПЕРЕДАЧИ СИГНАЛОВ НАВИГАЦИОННЫХ СПУТНИКОВ

Зайцев И.Д., Юрченко Е.С.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Герасимович В.Ю. – старший преподаватель

Рассмотрены базовые концепции передачи сигналов навигационных спутников и приведены примеры моделей в среде Matlab.

Для передачи сигналов навигационных спутников исходный сигнал, представленный нулями и единицами, должен подвергнуться модуляции. В большинстве случаев оказывается, что спектр исходного сигнала сосредоточен не на тех частотах, которые эффективно пропускает имеющийся канал. Поэтому необходимо преобразовать исходный сигнал таким образом, чтобы полученный сигнал соответствовал всем требованиям канала. Этот процесс и называется модуляцией, сущностью которого является формирование некоторого колебания, называемого несущим колебанием или просто несущей, и изменение какого-либо параметра колебания во времени пропорционально исходному сигналу. [2] Если в качестве несущего колебания использовать гармонические колебания, то будет существовать три параметра, которые можно использовать для представления исходных данных: амплитуда, частота и фаза.

В спутниковой навигации применяется бинарная фазовая манипуляция (BPSK – Binary Phase Shift Keying). Так как в цифровых системах существует только два состояния, модуляция будет заключаться в смещении фазы модулируемого сигнала на одно из двух значений (ноль или  $\pi(180^\circ)$ ). [1] При практической реализации выбирается значение частоты несущего колебания и происходит кодирование нулей и единиц через последовательные интервалы во времени  $T_b$  таким образом, что единица кодируется несущей частотой с нулевой фазой, а ноль – со сдвигом фазы в  $180^\circ$ . Стоит заметить, что исходный цифровой сигнал нужно модифицировать к интервалу  $-1 +1$  и только затем обрабатывать. Полученный BPSK сигнал можно рассматривать как произведение двух сигналов – несущего колебания и исходных данных, принимающих значения либо  $+1$ , либо  $-1$ . [3]

Таким образом с помощью BPSK происходит кодирование нулей и единиц. Однако возникает проблема интерференции и наложения сигналов друг на друга, потому что спутник передает одновременно множество сигналов. Для решения такой проблемы используется расширение спектра методом прямой последовательности (DSSS – Direct Sequence Spread Spectrum). Такой метод является расширением BPSK и характеризуется наличием третьего компонента, называемым сигналом распространения или PRN (PRN – Pseudo Random Noise) кодом. [3]

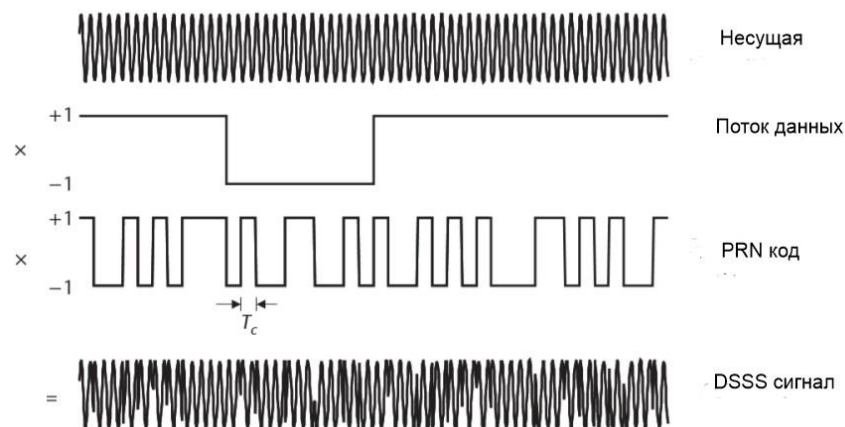


Рисунок 1 – DSSS модуляция

В общем случае навигационный космический аппарат(НКА) GPS излучает три PRN кода:

1. Точный P (Protected/Precise) код, который является основным навигационным кодом;
2. Y-код, используемый вместо P-кода, когда нужно исключить возможность несанкционированного использования P-кода;
3. C/A (Coarse/Acquisition) код грубого поиска, который используются в гражданских нуждах.

Для передачи информации в GPS применено кодовое разделение радиосигналов, т.е. все НКА используют одни и те же несущие частоты, но каждый из спутников имеет свои коды, свойства которых позволяют надежно (с заданной вероятностью) разделять сигналы различных НКА между собой.

C/A-код формируется с помощью двух сдвиговых регистров G1 и G2, имеющих по 10 разрядов,

и число символов (чипов) C/A-кодов составляет  $M_{C/A} = 2^{10} - 1 = 1023$ . Кодовая последовательность из 1023 символов формируется с тактовой частотой  $f_{\text{чип}} = 1,023$  МГц и имеет длительность 1 мс. Для формирования C/A кода используется алгоритм последовательности Голда.

P-код является основным более точным дальномерным кодом. Используется в военных целях.

Y-код представляет собой зашифрованный P-код с помощью системы A/S (Anti/Spoofing).

Таким образом, можно передавать различные сигналы на одной несущей частоте. Приемник заранее знает частоту несущей и PRN код. Фактически, PRN код является ключом при демодуляции сигнала приемником. Такая технология получила название множественного доступа с кодовым разделением (CDMA – Code Division Multiple Access). Ее идея заключается в передаче сигналов одновременно и на одной частоте, но каждый такой сигнал имеет свой PRN код. Каждый спутник имеет свой уникальный PRN код, который вшивается в исходный сигнал. Приемник знает PRN коды всех спутников и может определить от какого спутника был принят сигнал. Это обеспечивает разделение данных от разных спутников.

Таким образом, PRN код служит для идентификации передающего спутника. PRN коды обладают такими полезными свойствами, как:

1. Свойство ортогональности, т.е. сигналы, пришедшие с разных спутников, не интерферируют друг с другом.
2. Использование PRN кодов позволяет повысить помехоустойчивость канала передачи. [4]

На рисунках 2 и 3 приведены примеры моделирования CDMA технологии для одной несущей и одинаковых исходных данных, но с разными PRN кодами.

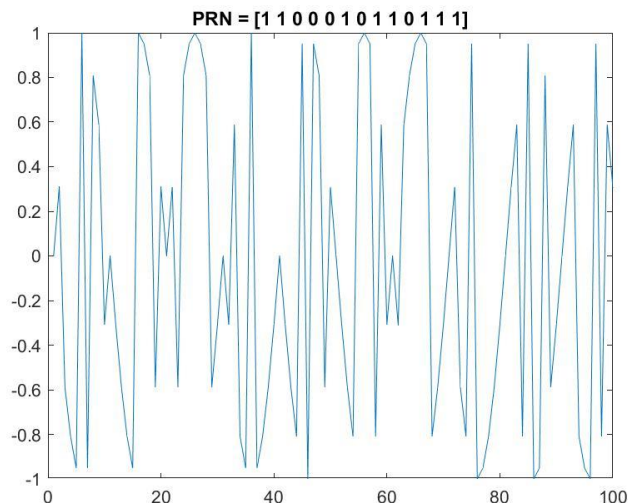


Рисунок 2 – Пример использования CDMA.

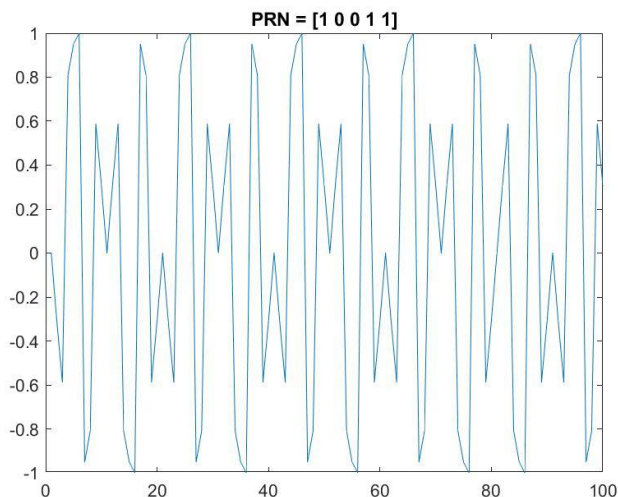


Рисунок 3 – Пример использования CDMA технологии.

**Список использованных источников:**

1. Скляр Б. Цифровая связь. Теоретические основы и практическое применение. Вильямс, 2003.
2. Сергиенко А.Б. Цифровая обработка сигналов: учеб. Пособие. БХВ-Петербург, 2011.
3. Kaplan E.D., Hegarty C.J. Understanding GPS: Principles and Applications. ARTECH HOUSE, 2006.
4. Власов В.М., Мактас Б.Я., Богумил В.Н. Беспроводные технологии на автомобильном транспорте. ИНФРА-М, 2018.

## ЭЛЛИПТИЧЕСКИЕ КРИВЫЕ В ЭЛЕКТРОННОЙ ЦИФРОВОЙ ПОДПИСИ

Коминч В.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Станкевич А.В. – к.т.н., доцент

В век технологий, когда электронный документооборот, если не преобладает над бумажным, то составляет ему большую конкуренцию, особенно важно задуматься об идентификации и сохранности важных данных. Множество предприятий, перешедших на электронную документацию, всячески пытаются обезопасить себя от подделок каких-либо ценных бумаг и попадания их «в руки» конкурентов. Ученые и специалисты всего мира для этих целей разрабатывают всё новые алгоритмы электронной цифровой подписи. В данном докладе рассматривается алгоритм электронной цифровой подписи на эллиптических кривых, описанном в белорусском стандарте СТБ 34.101.45 – 2013, а также его аппаратная реализация на базе ПЛИС.

Для работы алгоритма необходимы несколько параметров: входное сообщение, имеющее произвольную длину, личный ключ и параметры эллиптической кривой, необходимые для формирования подписи. Длина личного ключа зависит от уровня стойкости, который зависит от алгоритма хэширования, используемого в процессе генерации ЭЦП.

Уравнение эллиптической кривой над конечным полем  $F_p$  выглядит следующим образом:

$$y^2 = x^3 + a * x + b \pmod{p} \quad (1),$$

где  $p$  - большое простое число.

Оно представляет собой кривые Вейерштрасса, обладающие некоторыми свойствами. Кривая имеет характеристику 0, поэтому является плоской. Она не имеет самопересечений, что достигается выполнением условия дискриминанта.

Множество точек, являющихся решением данного уравнения, образуют конечное поле. В данном поле определена одна особая точка, расположенная на бесконечности и представляющая аналог нуля для чисел, и некоторые операции над точками кривой, такие как сложение и умножение точки на константу. Также определен обратный элемент, который получается вычитанием из ординаты координаты точки.

Операция сложения определена как алгебраически, так и геометрически. Геометрический смысл сложения заключается в пересечении прямой, образующейся двумя слагаемыми точками, с кривой и отражении полученной точки относительно вертикальной прямой. Алгебраическая форма имеет две формулы для равных и разных точек.

Операция умножения представляется через сложение точки  $N$  раз. Таким образом, напрямую операцию невозможно выполнить. В работе рассматриваются некоторые возможные реализации данной операции.

С операцией умножения на константу связана основная задача по взлому эллиптических кривых. Она носит название задачи дискретного логарифмирования. Её суть заключается в подборе того коэффициента, на который умножали базовую точку. Именно этот коэффициент является закрытым ключом.

Также определена операция умножения двух чисел по модулю. Так как числа имеют большую длину (минимально 128 бит), то для их реализации необходимо использовать некоторые схемы ускоренного умножения. Принципиальная разница заключается лишь в способе вычисления, так как возможно найти результат умножения, а затем найти остаток от деления, а можно проводить последовательные упрощения, сокращая разрядность промежуточного результата.

Кроме того, нетипичную реализацию имеет операция деления. Она определена таким образом, чтобы заменить деление умножением с взаимно простым числом. Данная операция представляет собой нахождение взаимно простых чисел.

Вместе с тем, большую роль выполняет реализация операции умножения на константу, так как она подвержена, так называемым, атакам по времени. В зависимости от значения личного ключа, время работы алгоритма может отличаться. Поэтому, чтобы обезопасить устройство, на практике алгоритмы «уравновешивают», делая атаки по времени сложно достижимыми.

На практике одним из самых распространенных алгоритмов является алгоритм Монтгомери, позволяющий выполнять умножение точки на константу за константное время. Другим способом безопасного вычисления является использование оконной функции, с подобранным размером окна.

На рисунке 1 представлена структурная схема устройства для генерации цифровой подписи. В случае использования эллиптической кривой из стандарта схему можно упростить, убрав генератор параметров эллиптической кривой и подавая на вход блоков электронной подписи соответствующие параметры. Кроме того генератор ключей также является необязательным блоком в схеме, так как

пользователь может иметь уже сгенерированный ключ. Сообщение для устройства подаётся блоками, так как изначально алгоритм может работать с данными произвольной длины.

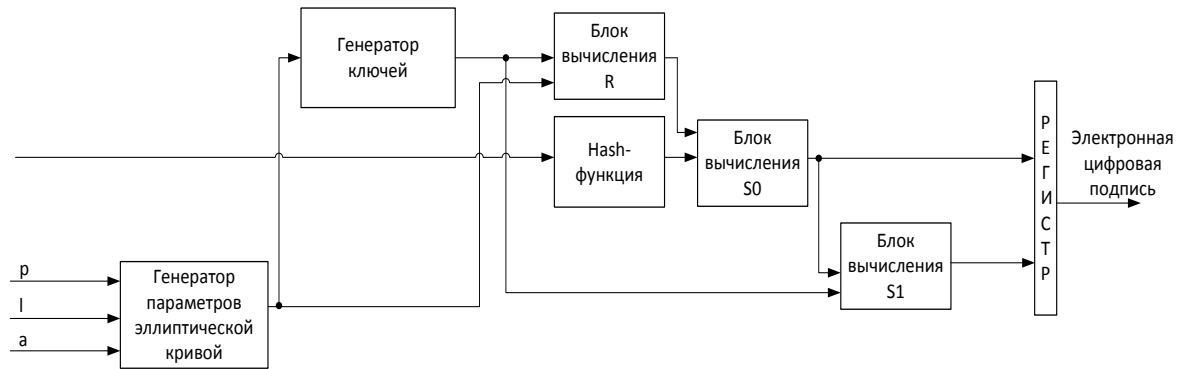


Рисунок 1 – Структурная схема устройства для выработки электронной цифровой подписи

$R = k * G$ , где  $G$  – базовая точка, являющаяся выходом генератора параметров эллиптической кривой. Блоки  $S_0$  и  $S_1$  являются частями итоговой подписи и выражаются следующими формулами:

$$S_0 = \langle \text{belt} - \text{hash}(\text{OID}(h)) \parallel \langle R \rangle_{2l} \parallel H \rangle_l,$$

$S_1 = \langle (k - \bar{H} - (\bar{S}_0 + 2^l)d \bmod q) \rangle_{2l}$ , где belt-hash – функция хэширования,  $H$  – выход блока хэш-функции,  $\text{OID}$  – идентификатор функции (06092A7000020022651F51<sub>16</sub>),  $R$  – выход блока вычисления  $R$ ,  $d$  – личный ключ,  $q$  – порядок группы точек из поля  $\mathbb{F}_q$ .

Идентификатор функции для алгоритмов, не представленных в стандарте, может быть вычислен по алгоритму из стандарта.

Уровень стойкости равен 128. Длина итоговой подписи равна 384 бита.

Одним из блоков алгоритма электронной цифровой подписи является блок вычисления функции хэширования. По умолчанию, в стандарте используется функция из стандарта СТБ 34.101.31-2011, в основе вычисления которой лежит алгоритм блочного шифрования из того же стандарта. При использовании данной хэш-функции в алгоритме электронной цифровой подписи на ее вход подается сообщение фиксированной длины. На выходе функции получается хэш-значение длиной 256 бит. Данный алгоритм используется в настоящей подписи. При выборе другого значения в качестве уровня стойкости данный алгоритм будет некорректным и необходимо выбрать другой.

Алгоритм шифрования, приведённый в стандарте, имеет множество архитектурных решений. В настоящей работе рассматриваются различные реализации и приводится их сравнение. Основными критериями сравнения были производительность и расход ресурсов.

Рассматриваются различные архитектурные решения операций в конечном поле, которые являются модулярными и должны удовлетворять временным ограничениям для невозможности реализации атаки по времени. Кроме того, они должны иметь приемлемый расход ресурсов для того, чтобы всё устройство можно было разместить в кристалле.

Также при реализации алгоритма рассматриваются несколько систем координат, упрощающих аппаратную реализацию либо усложняющих взлом подписи. Они, напрямую, влияют на реализацию операций по модулю, делая их сложнее, но избегая операции деления.

Наконец, для проверки правильности генерации подписи используется алгоритм проверки, приведённый в стандарте и реализованный аппаратно. Он включает в себя те же операции, что и устройство формирования подписи.

Сильной стороной устройств на основе эллиптических кривых несомненно защищённость информации от кражи. Вместе с тем, наличие у злоумышленника личного ключа пользователя

Приведенное устройство реализуется на базе FPGA семейства Virtex 7 фирмы Xilinx.

Аппаратную реализацию цифровой подписи можно использовать для проверки транзакций криптовалют, цифровых операций с валютами (платежами), подписи на электронные документы и любые другие файлы.

#### Список использованных источников:

1. Информационные технологии. Алгоритмы электронной цифровой подписи и транспорта ключа на основе эллиптических кривых : СТБ 34.101.45–2013. – Введ. 30.08.2013. – Минск : Госстандарт, 2013. – 41 с.
2. Информационные технологии. Криптографические алгоритмы шифрования и контроля целостности : СТБ 34.101.31–2011. – Введ. 31.01.2011. – Минск : Госстандарт, 2011. – 32 с.
3. [https://ru.wikipedia.org/wiki/ Эллиптическая\\_кривая](https://ru.wikipedia.org/wiki/Эллиптическая_кривая)
4. [https://ru.wikipedia.org/wiki/ Дискретное\\_логарифмирование\\_на\\_эллиптической\\_кривой](https://ru.wikipedia.org/wiki/Дискретное_логарифмирование_на_эллиптической_кривой)
5. [http://wiki.ru/wiki/Алгоритм\\_Монтгомери\(эллиптические\\_кривые\)](http://wiki.ru/wiki/Алгоритм_Монтгомери(эллиптические_кривые))

# ФОРМИРОВАНИЕ ДИАГРАММЫ НАПРАВЛЕННОСТИ ЦИФРОВОЙ ФАЗИРОВАННОЙ АНТЕННОЙ РЕШЕТКИ

Конопелько Я.Д., Егоров А.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Матюшков А.Л. – к.т.н., доцент

В работе рассматриваются основные принципы функционирования фазированных антенных решеток и методы формирования диаграммы направленности. Рассматриваются алгоритмы формирования лучей диаграммы направленности, коррекции межканального рассогласования приемных каналов в цифровой антенной решетке. А также решается задача построения алгоритма, структурной схемы и математической модели диаграммообразующей схемы РЛС кругового обзора.

Применение фазированных антенных решеток в радиолокационных системах имеет многолетнюю историю, однако развитие элементной базы позволяет расширить область их применения. Раньше создание таких систем требовало применение большого количества устройств, по этой причине ФАР использовались исключительно в больших стационарных радарх. Достижения в методах проектирования ФАР дали возможность широкого внедрения в различные радиотехнические системы и комплексы. Принцип действия фазированной антенной решетки представлен на примере линейной ФАР на рисунке 1.

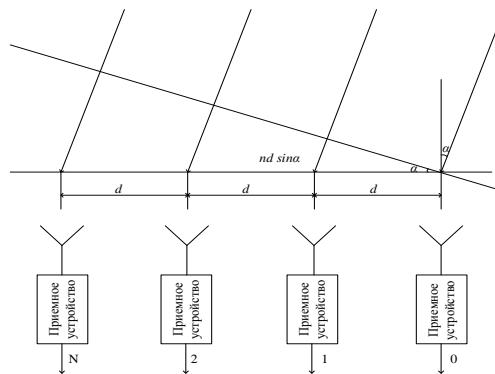


Рисунок 1 – Принцип действия линейной фазированной антенной решетки

Если расстояние между элементами равно  $d$  и фронт волны распространяется под углом  $\alpha$ , то при длине волны  $\lambda$  относительная фаза на элемент  $n$  равна  $(2\pi nd \sin(\alpha)/\lambda)$  [1]. Сигналы от  $N$  элементов складываются без какого-либо взвешивания и описываются формулой:

$$\dot{Y} = \sum_{n=1}^N \dot{X}_n e^{-i \frac{2\pi nd \sin \alpha}{\lambda}} \quad (1)$$

Из выражения (1) с учетом межканального рассогласования приемных каналов РЛС в общем виде формула формирования луча имеет вид:

$$\dot{B}_r = \sum_{n=1}^N \dot{A}_n \dot{P}_n \dot{C}_n \quad (2)$$

где  $\dot{A}_k$  – комплексный сигнал от  $n$ -ого элемента;  
 $\dot{P}_k$  – комплексная функция луча;  
 $\dot{C}_k$  – комплексный коэффициент коррекции амплитудно-фазовых характеристик антенной решетки.

При равномерном распределении лучей в секторе обзора по углу места  $S$  и шириной луча  $\beta$ , получим количество лучей  $k$  равное  $S/\beta$ . Для луча с максимумом  $\varepsilon_{max}$  с учетом относительной фазы на элементах ФАР комплексная функция луча представляет собой:

$$\dot{P}_k = |P_k| \left( \cos\left(\frac{2\pi \sin(\varepsilon_{max})d}{\lambda}\right) - i \sin\left(\frac{2\pi \sin(\varepsilon_{max})d}{\lambda}\right) \right) \quad (3)$$

где  $\varepsilon_{max}$  – максимум луча ДН  
 $|P_k|$  – амплитуда (при равно амплитудном суммировании равняется 1).

Стабильность параметров лучей диаграммы направленности, а следовательно, и точности измерения угла во многом зависят от идентичности амплитудно-фазовых характеристик приемного тракта. На практике сигнал при прохождении через приемный тракт РЛС подвергается различным дестабилизирующим факторам. Цифровой метод формирования ДН дает возможность компенсировать межканальное рассогласование приемных каналов [2]. Минимизация разброса между каналами производится путем соответствующей подстройки весового коэффициента, который приводит к минимальной ошибке  $\hat{\Delta}_{min}$ :

$$\hat{\Delta}_{min} = \hat{U}_0 - w\hat{U}_k \quad (4)$$

где  $\hat{U}_0$  – комплексная амплитуда сигнала на входе опорного канала;  
 $w$  – весовой поправочный коэффициент;  
 $\hat{U}_k$  – напряжение сигнала на выходе регулируемого сигнала

Комплексный весовой подстроечный коэффициент рассчитывается для всех  $N$  приемных каналов относительно одного из приемных каналов в режиме калибровки (на все приемные элементы подается равноамплитудный когерентный сигнал с частотой  $f$ ) и имеет следующий вид:

$$\hat{W}_k = \frac{\alpha_k \sum_{n=1}^N (x_0(n)x_k(n) - y_0(n)y_k(n))}{M} + j \frac{\alpha_k \sum_{n=1}^N (x_0(n)x_k(n) - y_0(n)y_k(n))}{M} \quad (5)$$

где  $k$  – номер канала;  
 $M$  – количество дискретов, используемых для вычисления весового коэффициента;  
 $\alpha_k$  – корректирующий множитель;  
 $x_0, y_0$  – действительная и мнимая части соответственно сигнала основного канала;  
 $x_k, y_k$  – действительная и мнимая части соответственно сигнала настраиваемого канала;

В результате моделирования, проведенного в среде MathWorks MATLAB для  $N=48$ ,  $S = 55^\circ$  и шириной луча  $\varepsilon_{0.5p} = 2.4^\circ$  получили диаграммы направленности для 23 лучей. Так же было проведено математическое моделирование алгоритма расчета весовых подстроечных коэффициентов позволяющих скорректировать амплитудно-фазовое рассогласование приемных каналов.

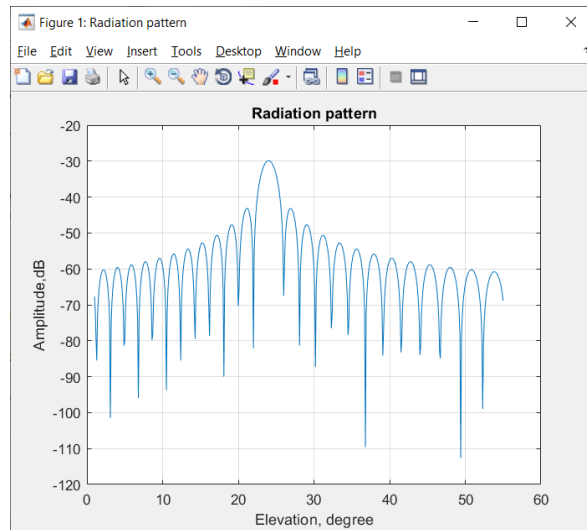


Рисунок 2 – Диаграмма направленности для луча  $r=10$  ( $\varepsilon_{max} = 24^\circ$ )

**Список использованных источников:**

1. Григорьев, Л.Н. Цифровое формирование диаграммы направленности в фазированных антенных решетках/ Л.Н. Григорьев. – М.: Радиотехника, 2010 – 144 с.
2. Активные фазированные антенные решетки/ Под ред. Д.И. Воскресенского и А.И. Канащенкова. – М.: Радиотехника, 2004. – 408с.

# ВЫСОКОПРОИЗВОДИТЕЛЬНАЯ РЕАЛИЗАЦИЯ НМАС SHA-256 НА БАЗЕ FPGA

Корбут А.А.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Станкевич А. В. – канд. технич. наук

Хеш-функции предназначены для создания «отпечатков» или «дайджестов» для сообщений произвольной длины. Применяются в различных приложениях или компонентах, связанных с защитой информации. НМАС – код проверки подлинности сообщений, используется для проверки целостности информации.

## Алгоритм SHA-256

Входное сообщение дополняется и делится на 512-битные блоки. В свою очередь каждый блок разделяется на шестнадцать 32-битных слов, после чего по формуле (1) генерируются 48 дополнительных слов[1].

$$\begin{aligned} s0 &= (w[i - 15] \gg 7) \oplus (w[i - 15] \gg 18) \oplus (w[i - 15] \text{ shr } 3) \\ s1 &= (w[i - 2] \gg 17) \oplus (w[i - 2] \gg 19) \oplus (w[i - 2] \text{ shr } 10) \\ w[i] &= w[i - 16] + s0 + w[i - 7] + s1 \end{aligned} \quad (1)$$

где shr – логический сдвиг вправо.

Обработка одного блока алгоритма выполняется за 64 итерации алгоритма. Схема одной итерации приведена на рисунке 1.

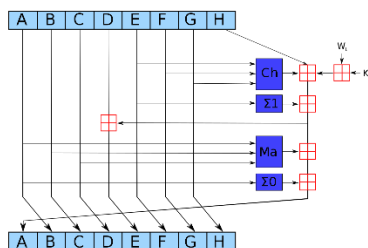


Рисунок 1 – Схема одной итерации алгоритма

A-H первоначально задаются константами. Впоследствии при обработке каждого последующего блока загружаются значения, вычисленными на предыдущем этапе. W – слово, k – константа, зависящая от номера итерации. Формулы для вычисления Ch,  $\Sigma 1$ , Ma,  $\Sigma 0$  вычисляются по формулам (2), (3), (4), (5) соответственно.

$$Ch = (E \& F) \oplus ((\text{not } E) \& G) \quad (2),$$

$$\Sigma 1 = (E \gg 6) \oplus (E \gg 11) \oplus (E \gg 25) \quad (3),$$

$$Ma = (A \& B) \oplus (A \& C) \oplus (B \& C) \quad (4),$$

$$\Sigma 0 = (A \gg 2) \oplus (A \gg 13) \oplus (A \gg 22) \quad (5),$$

После обработки 64-итерациями алгоритма выходные значения A-H складываются с первоначальными значениями A-H, загруженными на первом раунде цикла. Хеш-значение получается после обработки всех блоков сообщения, путем склеивания последних сумм A-H (6).

$$\text{hash} = h0 \parallel h1 \parallel h2 \parallel h3 \parallel h4 \parallel h5 \parallel h6 \parallel h7 \quad (6),$$

## Алгоритм НМАС

Реализуется путем суммирования по модулю 2 ключа с константой i\_pad, после чего аналогично складывается с входным сообщением, затем вычисляется первая hash-сумма. После этого ключ суммируется (сумма по модулю 2) с константой o\_pad, результат суммируется с hash-суммой 1 и вычисляется путем обработки алгоритмом sha-256 hash-sum2[1]. Пример показан на рисунке 2.

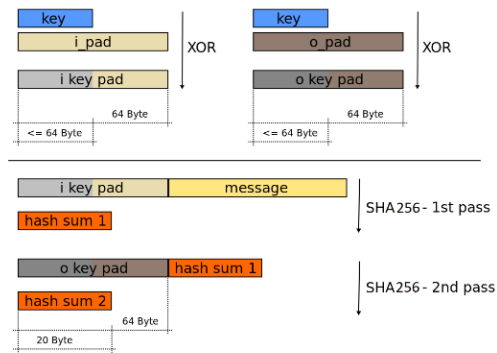


Рисунок 2 – Реализация HMAC-SHA256

Высокая производительность будет достигаться путем параллельно-итеративно-конвейерной реализации двух алгоритмов. Ключ для HMAC-алгоритма ограничим 512 битами, что позволит обеспечить достаточную надежность и уменьшит затраты ресурсов ПЛИС.

Значения ключа и сообщения будем накапливать в блочной памяти (BRAM). Для их приема задействуем общий 64-битный вход. Подачей на вход BRAM управляет демультиплексор. Внутри блока «Block divider» после накопления ключа в последующих тактах произойдет сложение с константой  $i\_pad$  и выдача 512-битного блока на обработку в экспандер. Одновременно с этим память будет продолжать накопление сообщения. После окончания приема сообщения, оно также будет разделено на 512-битные блоки и подано на обработку алгоритмом SHA-256. Первое вычисленная хеш-сумма вернется в блок HMAC для повторного суммирования с константой  $o\_pad$ , после чего будет произведено вычисление нового хеш-значения, являющимся окончательным. Описанная схема представлена на рисунке 3.

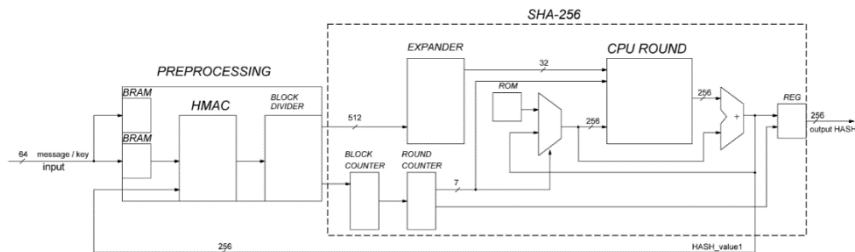


Рисунок 3 – Схема IP-ядра

Также была разработана ускоренная схема для расчета 64 раундов алгоритма SHA-256 с предварительным расчетом переменных[2]. Данная схема использует трехвходовой сумматор с ускоренным переносом, основанный на базовых элементах LUT6\_2 и CARRY\_4[3].

**Список использованных источников:**

1. D. Eastlake, T. Hansen, RFC 4634 US Secure Hash Algorithms (SHA and HMAC-SHA): Federal Information Processing Standard / AT&T Labs – USA, 2006.
2. An ASIC Design for a High Speed Implementation of the Hash Function SHA-256 / GLSVLSI'04// Boston, Massachusetts, USA., April 26–28, 2004. – P. 421-425.
3. Ternary Adder IP Cores / Martin Kumm, Jens Willkomm // 2013. – P. 1-4.

## ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ СИСТЕМ СПУТНИКОВОЙ НАВИГАЦИИ

**Краснов А.А., Лавренчук С.А., Нистюк А.С.**

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

**Герасимович В.Ю.**

В данной работе рассматриваются вопросы, связанные со спутниковой навигацией. Спутниковая навигация GPS уже давно является стандартом для создания систем позиционирования и активно применяется в различных навигационных устройствах. Эта статья посвящена тому, как с помощью спутников можно установить местоположение пользователя, какие есть системы координат и как они помогают в определении позиции.



Сегодня быстрыми темпами развиваются и совершенствуются технологии для навигации с помощью систем глобального позиционирования. GPS предоставляет информацию о местоположении, скорости и времени, что позволяет эффективно использовать его во многих приложениях в повседневной жизни. Основной принцип, лежащий в основе всей системы GPS, прост и давно используется для навигации и ориентирования: если есть точная информация о местоположении какого-либо ориентира и расстояние до него, то можно начертить окружность, на которой должна быть расположена точка положения объекта (рисунок 1).

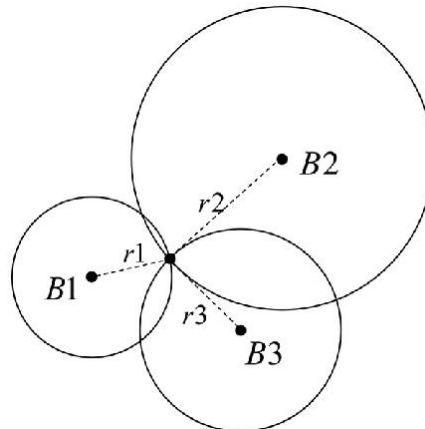


Рисунок 1 – 2D модель ранжирования TOA

GPS использует концепцию ранжирования TOA (Time of Arrival – TOA) для определения местоположения пользователя. Метод TOA основан на измерении задержки распространения радиосигнала между излучателем и приемником. Излучатель отправляет сигнал в точно известное для приемника время. Приемник измеряет временной интервал между отправкой сигнала излучателем и его получением. Этот временной интервал, называемый временем распространения сигнала, затем умножается на скорость сигнала (например, скорость звука или скорость света), чтобы получить расстояние от излучателя до приемника. Измеряя время распространения сигнала, передаваемого от нескольких излучателей (то есть навигационных средств) с известным местоположением, приемник может определить свои координаты. Минимальное количество измерений для определения местоположения пользователя равно трем.

Проводя измерения TOA для нескольких спутников, достигается трехмерное позиционирование. Метод аналогичен методу, описанному выше, за исключением того, что вместо окружностей, как показано на рисунке 1, используется пересечение сфер.

Для измерения и определения орбит спутников GPS удобно использовать Земле-центрированную инерциальную систему координат (Earth-Centered Inertial – ECI), в которой начало координат находится в центре масс Земли, а оси указывают в фиксированные направления относительно звезд. В данной системе координат предполагается, что  $xy$ -плоскость совпадает с экваториальной плоскостью Земли, ось  $+x$  постоянно зафиксирована в определенном направлении относительно небесной сферы, ось  $+z$  перпендикулярна  $xy$ -плоскости в направлении северного полюса, а ось  $+y$  выбрана так, чтобы сформировать правую систему координат.

Для расчета местоположения приемника GPS удобнее использовать систему координат, которая вращается вместе с Землей, известную как Земле-центрированная Земле-фиксированная система координат (Earth-Centered Earth-Fixed – ECEF). В такой системе координат проще вычислить параметры широты, долготы и высоты, которые отображает приемник. Также, как и Земле-центрированная инерциальная система координат (ECI), ECEF, используемая для GPS, имеет  $xy$ -плоскость, совпадающая с экваториальной плоскостью Земли. Однако в системе ECEF ось  $+x$  указывает в направлении долготы  $0^\circ$ , а ось  $+y$  указывает в направлении долготы  $90^\circ$ . В системе ECEF ось  $z$  выбирается перпендикулярно экваториальной плоскости в направлении географического северного полюса (то есть, где линии долготы встречаются в северном полушарии), тем самым завершая правую систему координат. Поэтому оси  $x$ ,  $y$  и  $z$  вращаются вместе с Землей и больше не описывают фиксированные направления в инерциальном пространстве.

Программное обеспечение для расчета орбит GPS включает преобразования между системами координат ECI и ECEF. Такие преобразования осуществляются путем применения матриц вращения к векторам положения и скорости спутника в системе координат ECI.

В результате процесса вычисления декартовы координаты  $(x, y, z)$  пользователя определяются в системе ECEF. Эти декартовы координаты преобразовываются в широту, долготу и высоту приемника. Однако, чтобы осуществить это преобразование, необходимо иметь физическую модель, описывающую Землю [1].

Стандартной физической моделью Земли, используемой для приложений GPS, является

Всемирная геодезическая система Министерства обороны 1984 года (World Geodetic System (WGS) 84). WGS 84 представляет эллипсоидальную модель формы Земли [2].

Как описано выше, пользователю требуется точная информация о позициях спутников GPS для определения положения, поэтому важно понимать, как вычисляются орбиты GPS. За вычисление орбиты отвечают Кеплеровы элементы орбиты. Кеплеровы элементы орбиты – это шесть параметров, определяющих положение небесного тела в пространстве. Первые два определяют форму орбиты, третий, четвертый и пятый – ориентацию плоскости орбиты по отношению к базовой системе координат, шестой – положение тела на орбите.

Спутниковые передатчики GPS используют метод прямой последовательности для расширения спектра (Direct Sequence Spread Spectrum – DSSS). Этот метод формирует широкополосный радиосигнал для передачи дальномерных сигналов и основных навигационных данных, таких как спутниковые координаты и состояние спутников. Дальномерные сигналы - это коды PRN (Pseudo Random Noise – PRN), которые с помощью двоичной фазовой манипуляции BPSK (Binary Phase Shift Key – BPSK) модулируют несущие частоты спутника. Эти коды выглядят и имеют спектральные свойства, схожие со случайными двоичными последовательностями, но на самом деле являются детерминированными.

**Список использованных источников:**

1. Understanding GPS: principles and applications/[editors], Elliott Kaplan, Christopher Hegarty.—2nd ed., // Artech house, 2006. – P. 723.
2. World Geodetic System [Электронный ресурс]: Википедия. Свободная энциклопедия. – URL: [https://en.wikipedia.org/wiki/World\\_Geodetic\\_System](https://en.wikipedia.org/wiki/World_Geodetic_System).

## **ДЕТЕКТОР ЗАИКАНИЯ НА ОСНОВЕ МАШИННОГО ОУБЧЕНИЯ**

*Лукашик Р.В.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Лихачёв Д.С. – доцент, канд. тех. наук*

Распознавание и классификация речевых дефектов, таких как заикание, является сложным процессом. Данная работа описывает процесс обработки речевого сигнала и выявление акустических параметров для распознавания и классификации заикания. Были сделаны выводы об использовании комбинации временных и частотных акустических параметров речи.

В настоящее время существует большое количество исследовательских работ, направленных на автоматическое распознавание речевых дефектов, таких как заикание. Большинство из них базируется на одном из трех классификаторах: искусственные нейронные сети, скрытые марковские модели и метод опорных векторов. Точность определения заикания, используя различные классификаторы, отличается у различных исследований из-за объема базы данных заикающей речи, использования искусственного или естественного заикания и др. параметров. Используя скрытые марковские модели исследователи добились 96% точности определения заикания [1].

При распознавании заикания стояла задача выбора акустических параметров речи, анализ которых должен быть дать наилучшие результаты в определении заикания.

Иногда при заикании возникают спонтанные мышечные спазмы речевого аппарата, которые блокируют произношение определенных звукосочетаний. Блокировку можно легко наблюдать, анализируя речевой сигнал во временной области, получая данные для статистического анализа. Анализируя речевой сигнал по временным параметрам, можно определить длительность сигнала, т.е. продолжительность звука.

Однако, анализируя речь только лишь по временным характеристикам, нельзя добиться высокой точности распознавания заикания. В этом случае может быть полезен анализ речевого сигнала в частотной области. Расчет мел-кепстральных характеристик (MFCC) может помочь в определении гласных звуков и обнаружении речевых нарушений, связанных с повторением гласных, а также их удлинением. Удлинение гласных является одной из значимых особенностей, характеризующих заикание. Из-за относительно большой продолжительности артикуляции гласных звуков их легче анализировать с использованием кепстрального анализа.

Повторение гласного или всего слога анализировать сложнее, чем продление. Это связано с тем, что продолжительность их артикуляции значительно короче, и по этой причине объем информации, получаемой из анализа отдельных фреймов значительно меньше. Однако эта проблема решается частотой встречаемости повторений, которая значительно выше, чем в случаях продления.

Исходя из этого были сделаны выводы о необходимости использования комбинации как временных, так и частотных акустических параметров речи для возможности более точного

определения повторяющихся фрагментов сигнала, характерных для заикания. Среди параметров речи были выбраны временные признаки: длительность сегмента и высота сегмента, а среди частотных признаков: частота основного тона, интенсивность, амплитуда, энергия.

В качестве базы для разрабатываемого инструментария извлечения акустических параметров была выбрана система OpenSMILE, в которой реализована функциональность первичной обработки речевого сигнала, т.е. погашение шумов за счет применения частотных фильтров, сегментация речевого сигнала с перекрытием, применение преобразования Фурье (FFT512) и извлечения MFCC.

Для анализа речевого сигнала были выбраны следующие параметры: длина фрейма – 1024 отсчета; число отсчетов в преобразовании Фурье – 512; окно Хэмминга; порядок сглаживания в косинус-преобразовании был установлен равным 12.

Важной и наиболее трудозатратой частью разработки детектора заикания является анализ речевых баз аудиозаписей людей с заиканием с целью выявления акустических параметров и обучения нейронной сети. Необходимо наработать базу аудиозаписей людей с заиканием различного возраста, пола, скорости речи и типа заикания: это может быть повторение отдельного слога, целого слова, запинки и другого рода заикания. База аудиозаписей должна быть достаточно обширной для возможности обучения нейронной сети, проверки корректности ее обучения и повышения точности распознавания речевых дефектов.

**Список использованных источников:**

1. Overview of Automatic Stuttering Recognition System, Lim Sin Chee, Ooi Chiam Sazali Yaacob, Universiti Malaisia Perlis, 02600 Jejawi Perlis, 2009
2. Jayanthi Sasisekaran, Effects of Delayed Auditory Feedback on Speech Kinematics in Fluent Speakers, University of Minnesota, 2012, PMID:23409597.
3. Marilia, Effect of delayed auditory feedback on stuttering with and without central auditory processing disorders, Brasil, 2017, PMID:29236907.

## **СИСТЕМА ОЦЕНКИ РЕЧЕВОЙ ФУНКЦИИ ДЛЯ ПАЦИЕНТОВ С БАС**

*Наркевич Е. С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Лихачев Д. С. – к.т.н., доцент*

На данный момент в Республике Беларусь наблюдается тенденция к росту инвалидности и смерти вследствие болезни двигательного нейрона, в частности самой распространенной ее разновидности – бокового амиотрофического склероза [1]. Разработка специализированного мобильного приложения для операционной системы Android ALS Expert, которое предоставляет функционал для записи, обработки и вывода результатов анализа голосовых тестов, является актуальным вопросом, поскольку данная разработка позволит врачам получить простой в использовании инструментарий для оценки состояния пациента. Для анализа голосового сигнала использована математическая модель на базе линейного дискриминантного анализа (ЛДА), для обучения которой использовалась база голосов, которая записана на базе Республиканского научно-практического центра неврологии и нейрохирургии.

Боковой амиотрофический склероз – заболевание, которое характеризуется поражением мотонейронов, развитием мышечных атрофий, бульбарных, дыхательных нарушений и быстрым течением заболевания [1, 2]. Разработка инструментария для ранней диагностики и оценки состояния пациентов является актуальной задачей для улучшения качества жизни пациентов, в том числе как средство сокращения времени, которое необходимо для постановки диагноза. Для оценки течения заболевания используется оценка нарушения голосовой функции пациента.

В качестве инструмента для диагностики, мониторинга и оценки состояния пациентов предлагается использовать мобильное приложение для операционной системы Android. Данный формат является наиболее удобным как для врачей, так и для пациентов, так как не требует специализированного оборудования и специально подготовленного помещения, в связи с повсеместностью использования смартфонов, также преимуществами данного формата является простота в установке и использовании.

Описываемое мобильное приложение ALS Expert предназначено для ранней диагностики БАС и отслеживания динамики прогрессирования заболевания на основе анализа нарушений речи. Основным функционалом приложения является запись голосового сигнала, его обработка и отображение общей оценки состояния пациента, а также акустических признаков, которые предназначены для интерпретации врачом. Голосовым тестом системы является длительное произношения звука [а].



Рисунок 1 – Архитектура системы оценки голосовой функции для пациентов с БАС

Анализ записанного голосового сигнала в соответствии с математической моделью осуществляется в вычислительном ядре приложения, которое реализовано на языке программирования C++. Для реализации пользовательского интерфейса и логики работы приложения используется специальный набор средств с открытым исходным кодом Flutter. Flutter является оптимальным решением для быстрой разработки кроссплатформенных приложений с удобным пользовательским интерфейсом. Для разработки Flutter-приложения используется язык программирования Dart [3]. Для оцифровки и сохранения голосового сигнала используется программный модуль, который реализован на языке программирования Java. Также данный модуль используется для связи между вычислительным ядром и модулем интерфейса. Описанная выше архитектура системы приведена на рисунке 1.

Для проведения теста пользователю необходимо нажать кнопку для записи голосового сигнала, после записи приложение производит анализ полученных данных. После обработки записи на экране отображаются следующие акустические признаки: джиттер, шиммер, индекс патологичности вибрато. Для удобства интерпретации данные выводятся в форме таблицы с вычисленными значениями и в форме нескольких гистограмм с отображением значения границы нормы признаков. После анализа нарушений речи пациента на основе полученных акустических признаков врач может сделать вывод о динамике состояния больного БАС.

В вычислительном ядре для обработки голосового сигнала использована математическая модель на базе линейного дискриминантного анализа (ЛДА). В качестве входных параметров модели используются следующие акустические признаки [4, 5, 6], которые извлекаются из записи голоса тестируемого человека:

1) джиттер – мера изменения частоты основного тона, которая является показателем стабильности работы фонаторной системы подсистемы речеобразования. Показатель растет при нарушении нейромоторной функции;

2) шиммер – мера вариативности амплитуды акустической волны во время фонации. Показатель растет при наличии речевых нарушений. Для оценки шиммера используется коэффициент амплитудных возмущений, который учитывает эффект "дрейфа" амплитуды сигнала, который возникает при естественном спаде интенсивности голоса;

3) энтропия периодов основного тона. Данный параметр используется для оценки нарушения способности контролировать стабильность частоты основного тона при протяжном произнесении звуков. Увеличение данного параметра говорит о превышении естественного уровня вариации основного тона;

4) степень патологии вибрато. Вибрато – это быстрое и регулярное колебание частоты основного тона во время длительной фонации. У больных БАС данный параметр характеризуется высокочастотными составляющими (около 9-14 Гц) [7].

5) среднеквадратичное отклонения частоты основного тона;

6) частотный диапазон фонации – параметр, характеризующий максимальное отношение частоты основного тона к минимальному.

7) направленный коэффициент возмущений.

В результате анализа голосового сигнала моделью вектор акустических признаков преобразуется в одну общую оценку, на основании которой определяется отношение голоса либо к категории здоровых, либо к категории больных. Если голос относится к группе больных, то исходя из

вычисленной с помощью ЛДА общей оценкой, отображается показатель степени нарушения голосовой функции, которая измеряется в шкале от 0 до 100 %. Для обучения математической модели использовалась записанная в Республиканском научно-практическом центре неврологии и нейрохирургии база голосов, которая состоит из 38 здоровых голосов и 25 голосов пациентов с БАС. Для базы голосов, как и для тестирования в приложении, использовалось длительное произношения звука [а] участниками записи.

Принимая во внимания тот факт, что продолжительность жизни пациентов с БАС составляет 3-5 лет, выживаемость каждого последующего года падает [1], а существующие средства диагностики ранней диагностики и последующего отслеживания развития заболевания имеют ряд недостатков и ограниченный функционал, разработка мобильного приложения для оценки голосовой функции для больных с БАС, в частности разрабатываемого приложения ALS Expert, позволит помочь врачам осуществлять раннюю диагностику боковой амиотрофического склероза и отследить развитие заболевания в динамике, так же позволит специалистам анализировать объективные параметры, что на данный момент недоступно при использовании аналогов.

**Список использованных источников:**

1. Рушкевич, Ю.Н. Уровень инвалидности вследствие болезни двигательного нейрона в Республике Беларусь / Ю.Н. Рушкевич, А.В. Копыток, С.А. Лихачев // Неврология и нейрохирургия. Вост. Европа. – 2017. – Т.7, №4. – С. 225–236.
2. Рушкевич, Ю.Н. Эпидемиология болезни двигательного нейрона в Республике Беларусь / Ю.Н. Рушкевич, С.А. Лихачев // Неврология и нейрохирургия. Вост. Европа. – 2018. – Т.8, №4. – С. 551–561.
3. Flutter – Beautiful native apps in record time [Электронный ресурс]. – Режим доступа: <https://flutter.dev/>. – Дата доступа: 10.04.2020.
4. Вашкевич М. И., Рушкевич Ю. Н. Детектирование речевых нарушений у пациентов с БАС на основе анализа голосового сигнала // Цифровая обработка сигналов. – 2019. – №4. – с. 33-41
5. Гвоздович А. Д., Рушкевич Ю. Н., Вашкевич М. И. Детектирование бульбарных нарушений при боковом амиотрофическом склерозе на основе анализа речевого сигнала // Доклады БГУИР. – 2018. – №. 6 (116) – С. 52-58
6. Клинический опыт применения частотно-временного анализа речевых сигналов в диагностике и мониторинге бульбарной дисфункции / Ю. Н. Рушкевич и др. // Неврология и нейрохирургия. Восточная Европа. – 2017. – Т.7, №3. – С.429 – 439.
7. Vashkevich M., Petrovsky A., Rushkevich Y. Bulbar ALS Detection Based on Analysis of Voice Perturbation and Vibrato // 2019 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA). – P. 267-272.

## **ИСПОЛЬЗОВАНИЕ ИНФОРМАЦИИ О ВОЗРАСТНЫХ ИЗМЕНЕНИЯХ СЛУХА ПРИ МОДЕЛИРОВАНИИ НЕЙРОСЕНСОРНОЙ ТУГОУХОСТИ**

*Порхун М.И.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Вашкевич М.И. – канд. тех. наук*

Предлагается подход к использованию информации о возрастных изменениях в слуховой системе человека при моделировании нейросенсорной тугоухости. Данный подход предлагается использовать в модели потери слуха, построенной на базе банка фильтров с применением компрессоров динамического диапазона в субполосах. Для настройки компрессоров модели используется информация об усреднённых порогах слышимости людей разных возрастов. В работе приведен пример моделирования нейросенсорной тугоухости для возраста 75 лет.

Для моделирования нейросенсорной тугоухости разработано множество методов. Однако, на практике, основными являются методы, основанные на субполосной декомпозиции звукового сигнала с последующим применением компрессоров динамического диапазона (КДД) в субполосах. Декомпозиция сигнала выполняется банком фильтров, согласованным с работой человеческого уха, а КДД выполняют задачу частотно-зависимого ослабления субполосного сигнала. Настройка параметров КДД, как правило, выполняется на основании аудиограммы тугоухого человека [1]. В большинстве случаев моделирование нейросенсорной тугоухости выполняется по аудиограмме слабослышащего человека относительно нормального слуха (абсолютного порога слышимости). Т.е. предполагается, что прослушивать результирующий сигнал будет человек с нормальным слухом. Нужно учесть, что целью моделирования является предоставление возможности человеку с нормальным слухом ощутить, как воспринимает звуки тугоухий человек. Для более точного достижения этой цели необходимо знать аудиограмму того человека («целевого» слушателя), который будет прослушать результирующий сигнал. Однако, как правило информация о слухе «целевого» слушателя неизвестна. В этом случае для подстройки модели можно использовать информацию об усреднённых порогах слышимости для разных возрастных групп. В данном случае предполагается, что возраст «целевого» слушателя известен.

Возрастное ухудшение слуха обусловлено как биологическими процессами старения организма, так и результатом воздействия шумов [2]. Наиболее острый слух наблюдается у людей в

возрасте от 14 до 19 лет. По мере старения человека абсолютный порог слышимости увеличивается, особенно в области высоких частот. Помимо того, с возрастом, отмечается ограничение верхнего порога воспринимаемых частот. При этом процесс ухудшения слуха с возрастом необратим [3].

В данной работе для моделирования нейросенсорной тугоухости с учётом возрастных изменений слуха предлагается использовать модификацию метода, описанного в [1]. Структура метода представлена на рисунке 1.

Для учёта возрастных изменений слуха предлагается использовать усреднённые пороги слышимости для разных возрастов (рисунок 2).

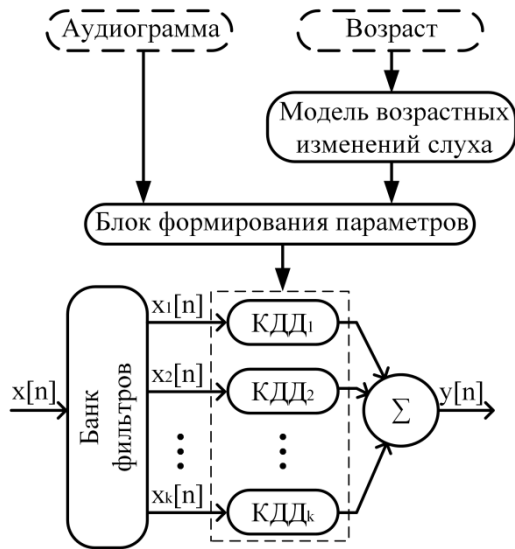


Рисунок 1 – Структура метода моделирования нейросенсорной тугоухости

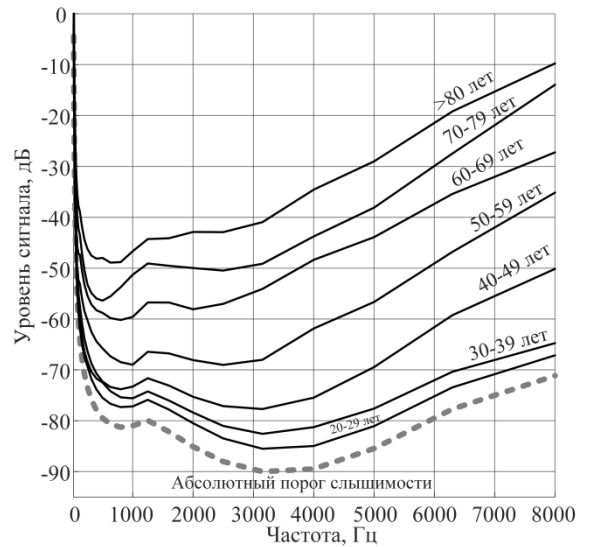


Рисунок 2 – Усреднённые пороги слышимости для людей разных возрастов

Входной сигнал  $x[n]$  раскладывается на субполосы при помощи банка гамматон-фильтров, имитирующий механизм частотного восприятия звука в ухе. Результирующие субполосные компоненты сигнала проходят через соответствующие компрессоры динамического диапазона (КДД), задачей которых является автоматический выбор коэффициентов ослабления субполос в зависимости от уровня мощности сигнала. Модель возрастных изменений слуха позволяет определить порог слышимости для конкретной частоты с учётом возраста человека без использования аудиограммы. Блок формирования параметров позволяет задать характеристику вход/выход для каждого КДД на основании либо аудиограммы тугоухого человека (если она известна), либо его возраста (если аудиограмма не известна). Выходной сигнал  $y[n]$  синтезируется путём суммирования обработанных КДД субполосных компонент.

Согласно [1], настройка модели нейросенсорной потери слуха выполняется на основании значений порогов слышимости на центральных частотах банка фильтров. На первом этапе определяются пороги слышимости тугоухого человека. Для этого на сетку центральных частот банка фильтров выполняется интерполяция либо аудиограммы «целевого» слушателя (если она известна), либо усреднённая аудиограмма для его возраста. На следующем этапе определяются пороги слышимости для тугоухого человека и слуха в норме. Исходя из значений полученных порогов слышимости, выполняется настройка характеристики вход/выход КДД. Пример зависимости порогов слышимости среди разных возрастов для частоты 1 кГц приведены на рисунке 3.

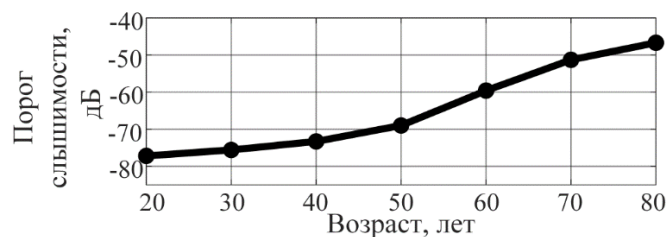


Рисунок 3 – Зависимость порогов слышимости от возраста для частоты 1 кГц

Проверка предлагаемого метода моделирования нейросенсорной тугоухости с учётом возрастных изменений слуха производилась на речевом сигнале (рисунок 4, а) для человека в возрасте 75 лет. Спектрограмма выходного сигнала (рисунок 4, б) отражает особенности восприятия

исходного звукового сигнала пожилым человеком.

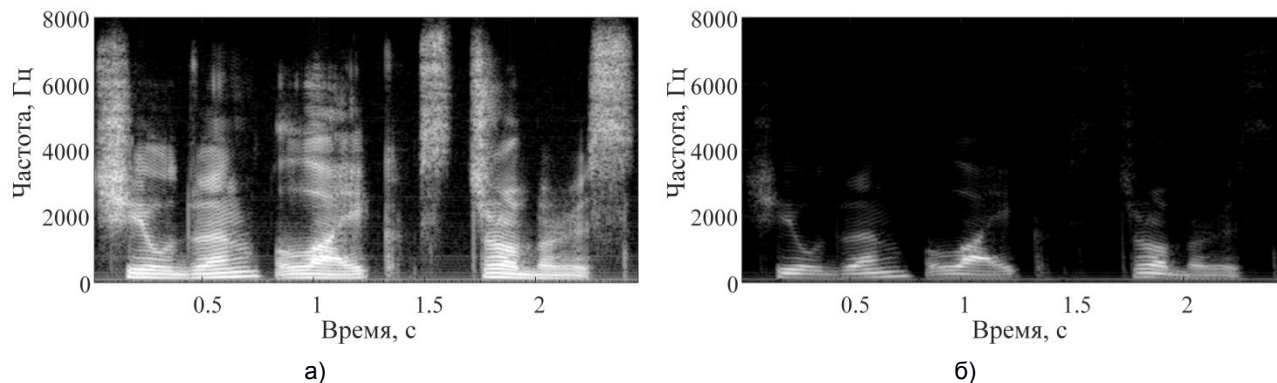


Рисунок 4 – Результаты обработки звукового сигнала предложенным методом: а) исходный звуковой сигнал; б) обработанный сигнал для возраста 75 лет

В работе предложен подход к использованию информации о возрастных изменениях слуха при моделировании нейросенсорной тугоухости. Ключевой особенностью метода является возможность моделировать возрастные слуховые патологии без использования аудиограмм конкретных людей. Работоспособность метода подтверждена результатами MATLAB-моделирования.

**Список использованных источников:**

1. Порхун, М. И. Моделирование эффекта потери слуха / М. И. Порхун, М. И. Вашкевич // Цифровая обработка сигналов и ее применение: труды 20-й междунар. конф., Москва, 28-30 марта 2018 г. / РНТОРЭС имени А.С. Попова. – Москва, 2018. – Т. 1. – С. 228 – 233.
2. Ярыгин В.Н. Руководство по геронтологии и гериатрии / В.Н. Ярыгин, А.С. Мелентьев. – М. : ГЭОТАР-Медиа, 2008. – Т. 4. – 528 с.
3. Старческая тугоухость (пресбиакузис) / А. И. Лопотко [и др.] // А. : Ылым, 1986. – 300 с.

## ХАРАКТЕРИСТИКИ И ВИДЫ ВСТРАИВАЕМЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

*Прокопеня А.С.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь)*

*Прокопеня А.С. – аспирант*

### Аннотация

В современном мире влияние встраиваемых вычислительных систем широко распространено. Встраиваемые вычислительные системы применяются в различных видах промышленности, а также в быту. На сегодняшний день существует большое количество различных вычислительных систем, о которых далее пойдет речь.

Цель статьи – дать краткие характеристики и виды встраиваемых вычислительных систем.

Влияние информационно-управляющих и встраиваемых систем широко распространено на сегодняшний день. Встраиваемые КС (компьютерные системы) широко используются в стиральных и посудомоечных машинах, холодильниках, телевизорах, автомобилях, в средствах коммуникации, умных приставках, а также в различных видах промышленности и т.д.

Различные виды задач автоматизации и способы их решения порождают все больше и больше вариантов встраиваемых вычислительных систем (ВВС).

Встраиваемые вычислительные системы – это специализированные вычислительные системы (ВС), которые взаимодействуют с объектом контроля или управления и объединены с ним единой конструкцией.

Встроенные системы разделяют на категории:

- Системы измерения.
- Системы автоматического управления (САУ).
- Системы передачи данных.
- Системы управления объектами.
- Автопилоты самолетов, автомобилей, морских судов.
- Мобильных телефонов.

- Специального медицинского оборудования.
- Игровых приставок.
- Цифровых музыкальных инструментов.
- Систем управления оружием.
- Сетевого оборудования.

Диапазон реализаций ВВС велик. В диапазон входят и устройства уровня таймера и более сложные иерархические системы, которые управляют объектами. В процессе разработки встроенной вычислительной системы, разработчик создает ВС, независимо от наличия готовых решений. Характеристики встраиваемых вычислительных систем делят на глобальные и локальные. К глобальным характеристикам относят:

- Производительность.
- Надежность.
- Оперативность.
- Стоимость.

В свою очередь к производительность делится на номинальную, комплексную, системную.

Локальные характеристики, определяют свойства отдельных устройств, частей системы, которые позволяют получить детальное представление о функционировании системы. Часто локальные характеристики являются частью математических выражений для расчета глобальных характеристик.

В настоящее время встраиваемые вычислительные системы широко распространены. ВВС делятся на определенные категории, а также имеют ряд характеристик. Встраиваемые вычислительные системы оказывают значительное влияние на сферу промышленности, а также оказывают значительное влияние на сферу жизнедеятельности человека.

**Список использованных источников:**

1. Ключев А.О., программное обеспечение встроенных вычислительных систем: учеб. пособие / Ключев А.О., П.В. Кустарев П.В., Ковязина Д.Р., Петров Е.В. – Слб: ИТМО: 2009 – 2013 с.
2. Комаровский, О. А. Встраиваемые вычислительные системы реального времени / О. А. Комаровский, И. А. Шимановский // Инновационные технологии в учебном процессе: материалы 51-й научной конференции аспирантов, магистрантов и студентов. (Минск, 17 апреля 2015 г.). – Минск: БГУИР, 2015. – С. 23 – 24.
3. Ozlib [Электронный ресурс]. – Режим доступа: [https://ozlib.com/847102/informatika/harakteristiki\\_vychislitelnyh\\_sistem](https://ozlib.com/847102/informatika/harakteristiki_vychislitelnyh_sistem). – Дата доступа: 01.04.2020.

## **АППАРАТНАЯ РЕАЛИЗАЦИЯ АЛГОРИТМОВ ДЕКОМПОЗИЦИИ БУЛЕВЫХ ФУНКЦИЙ**

*Самусенко М.А.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Бибило П.Н. – проф.*

Проектирование вычислительных и управляющих дискретных устройств тесно связано с решением разнообразных логико-комбинаторных задач, возникающих на разных этапах проектирования. К этим задачам относятся разработка и верификация алгоритмов логического управления, обеспечение их корректности и реализация на автоматном уровне, а также многие другие задачи, в частности – синтез логических схем, зачастую основанный на декомпозиции булевых функций. В настоящее время интерес к декомпозиции возрастает в связи с широким распространением программируемых логических интегральных схем (ПЛИС) типа FPGA (Field-Programmable Gate Array), основу которых составляют матрицы программируемых логических элементов, каждый из которых способен реализовать любую булеву функцию из ограниченного числа элементов..

Проблема декомпозиции булевых функций (другими словами, представления функций в виде суперпозиции (композиции) некоторых более простых функций) возникла при синтезе комбинационных логических схем. Так как булевы функции являются моделями логических элементов комбинационной схемы, то моделью всей схемы в целом является суперпозиция функций [1]. Получение по заданным булевым функциям тех или иных суперпозиций – функциональных разложений – и является целью декомпозиции. Декомпозиционный подход к синтезу многоуровневых схем является наиболее общим.

Обычно под декомпозицией понимается двухуровневая декомпозиция, тогда процесс получения многоуровневых логических схем может быть представлен в виде последовательности двухуровневых декомпозиций. Декомпозиция может применяться при синтезе и как способ предварительной технологически независимой оптимизации, и непосредственно на этапе технологического отображения, т.е. установки в логическую схему элементов используемого базиса синтеза [2].



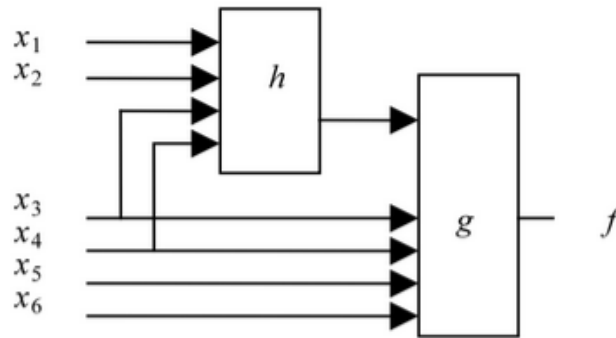


Рис.1 – Пример неразделительной последовательной двухблочной композиции

Основной задачей является изучение и оценка эффективности основных алгоритмов, используемых при декомпозиции булевых функций: декомпозиций «от входов», задач декомпозиции, сведенных к решению логических уравнений. Будут протестированы различные форматы задания булевых функций и проведена оценка зависимости сложности аппаратной реализации от числа аргументов функции. Сама разработка предполагает использование ее в виде IP-блока при создании специализированной цифровой аппаратуры.

**Список использованных источников:**

1. Бибило, П.Н. Декомпозиция булевых функций на основании решения логических уравнений / П.Н. Бибило. - Минск : Беларус. навука, 2009. – 211 с.
2. Закревский А.Д. Вычисления в многомерном булевом пространстве / А.Д. Закревский. – Минск : ОИПИ НАН Беларуси, 2011. – 106 с.

## ПРИМЕНЕНИЕ МОДЕЛИ ЦВИКЕРА ДЛЯ ОЦЕНКИ УРОВНЯ ГРОМКОСТИ ЗВУКОВОГО СИГНАЛА ДЛЯ ТУГОУХОГО ЧЕЛОВЕКА

Санько Н.С.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Вашкевич М.И. – к.т.н., доцент

Люди с особенностями слуха зачастую используют слуховой аппарат для более удобного взаимодействия с окружающим миром. Однако в слуховых аппаратах использование динамической компрессии сигнала иногда приводит к тому, что при обработке широкополосного сигнала, оцененная энергия каждой субполосы и восстановление сигнала до уровня, комфортного для восприятия, приводит к переусилению сигнала и громкость после обработки является достаточно высокой, что приводит к дискомфортному восприятию звука. Решением данной проблемы является более сложная модель оценки громкости сигнала для тугоухого человека.

Цель данной работы – модифицировать модель оценки громкости Цвикера и адаптировать её под слуховые характеристики человека с тугоухостью.

Основной идеей предлагаемого метода является оценка и сравнение уровней громкости при нормальном и при нарушенном слуховом восприятии, и на основе полученных оценок рассчитать коэффициенты усиления сигнала для восстановления до уровня, который бы был комфортен для восприятия человеку с тугоухостью.

Предлагаемый метод оценки громкости для людей с особенностями слуха разделен на этапы. В первом этапе на вход банка фильтров подаётся звуковой сигнал, после прохождения сигнала через банк, он поступает на две модели оценки громкости Цвикера (второй этап). Первая модель - для людей с особенностями слуха, вторая – для людей с нормальным слуховым восприятием. В третьем этапе производится расчёт коэффициентов усиления на основании сравненных громкостей. И в завершающем этапе рассчитывается выходной звуковой сигнал с учётом порогов восприятия человека с особенностями слуха.

На рисунке 1 представлена структурная схема модели оценки громкости для тугоухого человека.

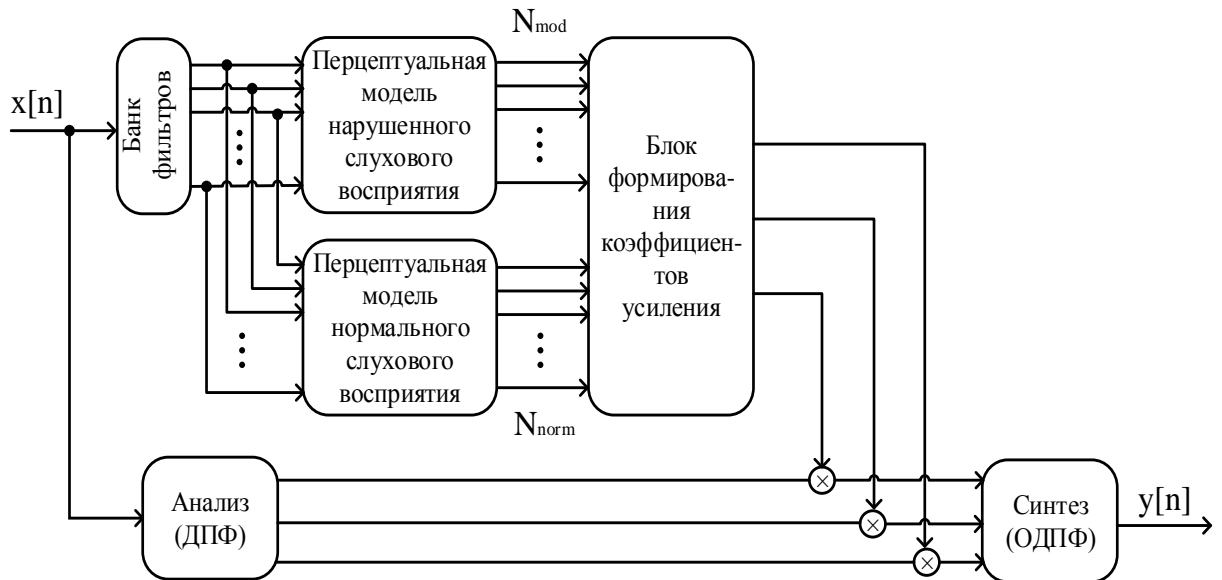


Рис. 1 - Структура модели Цвикера для оценки громкости звукового сигнала для тугоухого человека

В модели оценки громкости Цвикера для стационарного звука в качестве основы используется уровень возбуждения в зависимости от выбранной критической полосы. Общая громкость сигнала выражается путём интегрирования удельной громкости по всему диапазону. Высота звука в данной модели выражается в Барках, а измерение громкости в Сонах, либо Фонах.

Удельная громкость – громкость звука, измеренная на избранной критической полосе (Барк). Для определения зависимости данной составляющей используется закон Стивенса [1], из которого следует, что ощущение интенсивности увеличивается с ростом физической интенсивности согласно степенному закону. Это означает, что относительное изменение громкости пропорционально относительному изменению интенсивности. В модели Цвикера вместо громкости используется удельная громкость на каждой критической полосе (т.к. полная громкость рассчитывается путём интегрирования по всему диапазону), а вместо интенсивности используется уровень возбуждения, так как данная составляющая более точно описывает избирательность слуха, чем интенсивность звука. Закон Стивенса представлен в формуле 1:

$$\frac{N'}{\Delta N'} = k \frac{\Delta LE}{LE} \quad (1),$$

где  $N'$  – удельная громкость,  $\Delta N'$  – изменение удельной громкости,  $\Delta LE$  – изменение уровня возбуждения,  $LE$  – уровень возбуждения,  $k$  – постоянная показателя степени. Данная формула отображает зависимость удельной громкости, как величины ощущения, от уровня возбуждения, при возрастании которого, ощущение громкости увеличивается в степенной форме. Это говорит о том, что при росте уровня возбуждения на какой-либо из критических полос слуха, общий уровень ощущения громкости может заметно измениться, согласно степенному закону.

Корректировка уровней возбуждения  $LE$  с учётом передаточных характеристик уха происходит перед расчётом уровней удельной громкости [2], так как порог возбуждения в тишине не учитывает передаточные характеристики уха. Корректировка уровней производится по формуле 2:

$$LE_{\text{corr}} = LE - A_0 \quad (2),$$

где  $A_0$  – коэффициенты коррекции.

Математическое представление удельной громкости приведено в формуле 3:

$$N' = 0.0635 \times 10^{0.025 \times LTQ} \times \left[ (1-s + s^{0.1 \times (LE_{\text{corr}} - LTQ)})^{0.25} - 1 \right] \quad (3),$$

где  $LTQ$  – порог возбуждения в тишине без учёта характеристик передачи уха,  $s$  – пороговый коэффициент.

На рисунке 2 представлен порог восприятия без корректировки ( $LTQ$ ).

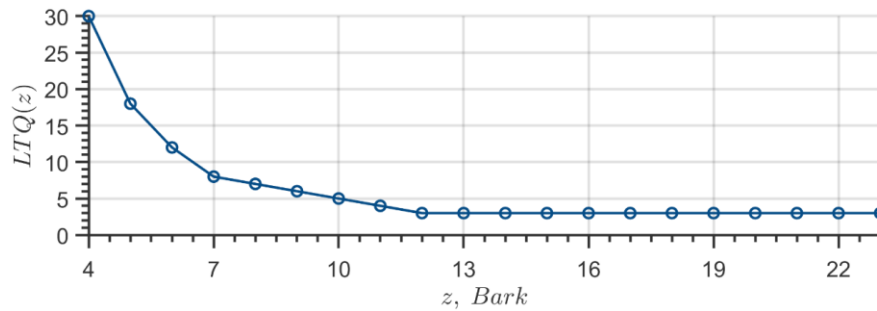


Рис. 2 – Порог восприятия слуха в тишине

На рисунке 3 представлен порог восприятия с учётом использования коэффициентов коррекции. ( $LTQ$  с коррективкой  $AO$ ).

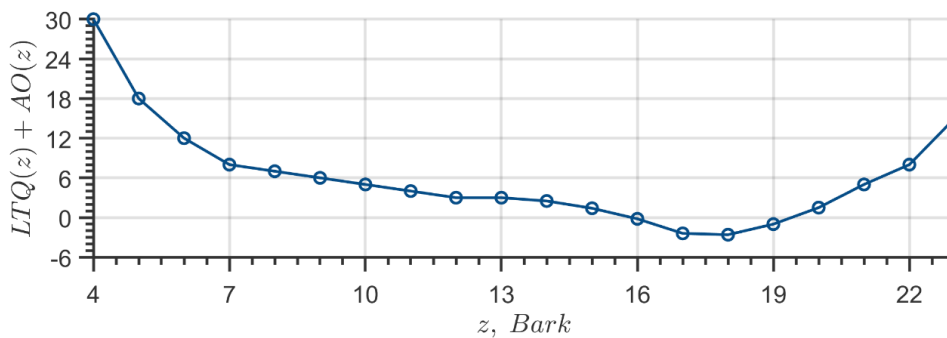


Рис. 3 – Порог восприятия слуха с учётом передаточных характеристик уха

Вывод: имея возможность смещения порогов слухового восприятия, существует возможность усовершенствования исходной модели оценки громкости Цвикера, при которой будет возможно оценивать громкость звукового сигнала и корректировать его для людей с особенностями слуха.

Список использованных источников:

1. Numerical Calculation and Experimental Research on Sound Loudness in Sound Field of Structural-Acoustic Coupling Cavity / J. Yuan [et al.] // *Technical Gazette*, 2014. – P. 1361-1369.
2. Implementing loudness models in Matlab / J. Timoney [et al.] // *Proc. of the 7<sup>th</sup> Int. Conference on Digital Audio Effects*, 2004.

## АНАЛИЗ АУДИОСИГНАЛОВ С ПОМОЩЬЮ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ НА БАЗЕ МОБИЛЬНЫХ ПЛАТФОРМ

Тарасевич В.В.

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Лихачёв Д.С. – канд. техн. наук

В данной работе описывается детектор правильности выполнения речевых заданий для систем коррекции речи, выполненный с использованием методов машинного обучения, его место в улучшении качества детектора речевых нарушений у людей с боковым амиотрофическим склерозом, а также краткий обзор используемых инструментов.

Системы обработки аудиосигналов всё чаще реализуются на основе мобильных платформ. Это системы для распознавания и поиска музыки, такие как “Яндекс.Алиса”, “Shazam”, “SoundHound”, приложения для понимания и обработки устной речи, такие как “Яндекс.Алиса”, “Google Assistant”, “Cortana” от компании Майкрософт и многие другие.

Среди таких систем есть и системы диагностики, в частности, системы диагностики Бокового Амиотрофического Склероза (БАС) [1]. БАС – неизлечимое нейродегенеративное заболевание. Основным методом диагностики данного заболевания занимает около года [1-2].

В исследовании [1] приводится описание системы автоматического детектирования речевых нарушений у пациентов с БАС при помощи методов машинного обучения. Суть метода заключается в акустическом анализе протяжного произношения звука [a], выделении ключевых выбранных признаков, их дальнейшей обработки классическими методами машинного обучения (методом k-

ближайших соседей и методом линейного дискриминантного анализа).

Методика детектирования, описанная в исследовании [1] опирается на тот факт, что произношение звука [a] является правильным, да и на то, что пациент произносит именно звук [a], а не какой-нибудь другой.

Целью данной работы является усовершенствование приложения созданного и описанного в [1-2] для автоматического детектирования речевых нарушений для диагностирования БАС при помощи оценки качества правильности произношения речевого задания (произношения звука [a]).

Для целей детектирования правильности произношения речевого задания хорошо подходят методы машинного обучения. Для распознавания речи много эффективных методов на основе машинного обучения, а распознавание протяжной буквы [a] – это по сути и есть распознавание фонемы, звука.

Используя методы машинного обучения необходимо определиться с признаками, которые будут выделяться для их последующей обработки этими методами. Признаки должны учитывать временную составляющую речи, а также частотную и амплитудную.

Для временной составляющей используется длительность фрагментов слитной речи. Также в работе используются пертурбационные признаки (амплитудные и частотные): джиттер, шиммер.

Для извлечения перечисленных признаков из аудиосигнала в данном исследовании используется проект OpenSMILE.

Проект OpenSMILE предназначен для автоматического извлечения признаков из аудиосигнала. SMILE в названии расшифровывается как “Speech and Music Interpretation by Large-space Extraction” что переводится, как “Интерпретация речи и музыки посредством многоуровневого выделения”. Детальное описание подходов используемых в OpenSMILE представлено в [3].

OpenSMILE используется для исследований и в приложениях для автоматической обработки речи и музыки в режиме реального времени.

Для обработки извлечённых признаков взят метод опорных векторов [4], поскольку он хорошо зарекомендовал себя на небольших обучающих выборках, что очень весомо для данной работы, так как имеющаяся выборка данных небольшая.

Предварительные эксперименты показали перспективность описываемого подхода.

#### **Список использованных источников:**

1. Вашкевич М.И., Рушкевич Ю.Н. Система детектирования речевых нарушений у пациентов с боковым амиотрофическим склерозом на основе теста на протяжное произнесение звука [a] // Актуальные проблемы неврологии и нейрохирургии. Рецензируемый сборник научных трудов. Выпуск 22. Под ред. д.м.н. Р.Р. Сидоровича и д.м.н., проф. С.А. Лихачева. – С. 278–290.
2. Detection of bulbar ALS using a comprehensive speech assessment battery/ Y. Yunusova [et al.] // Proceedings of the International Workshop on Models and Analysis of Vocal Emissions for Biomedical Applications – 2013. – P. 217-220
3. B. Schuller, B. Vlasenko, F. Eyben, M. Wöllmer, A. Stuhlsatz, A. Wendemuth, G. Rigoll. Cross-Corpus Acoustic Emotion Recognition: Variances and Strategies.
4. Nello Cristianini, John Shawe-Taylor. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. — Cambridge University Press, 2000.

## **МОДУЛЬ ЗВУКОВЫХ ЭФФЕКТОВ ДЛЯ СТРУННЫХ ИНСТРУМЕНТОВ НА БАЗЕ FPGA**

*Шарков Ф.Ф.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Герасимович В.Ю*

В данной работе будут рассмотрены некоторые цифровые аудиоэффекты. В качестве примеров были выбраны самые распространенные на сегодняшний день: задержка, эхо, overdrive, искажение (distortion). Будут рассмотрены их особенности, свойства, некоторые сходства и отличия.

В последнее время все большее количество музыкантов переходят с ламповых усилителей на цифровые устройства, ядром которых служит специализированный процессор. Несомненным преимуществом такого подхода является то, что такие устройства позволяют воспроизводить любые эффекты. Рассмотрим некоторые из них.

Суть эффекта задержка заключается в микшировании входного сигнала и его копию, задержанную на некоторую фиксированную величину.

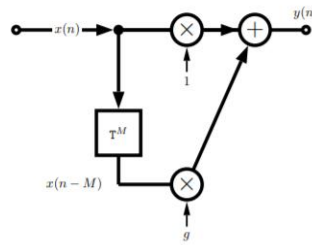


Рисунок 1 – Схема алгоритма работы устройства эффекта задержки

Слегка усложнив этот алгоритм, можно получить эффект эхо. Схематично алгоритм такого устройства представлен на рисунке 2. Коэффициент BL отвечает за усиление входного сигнала, FF – за усиление задержанного сигнала, FB – коэффициент реверберации.

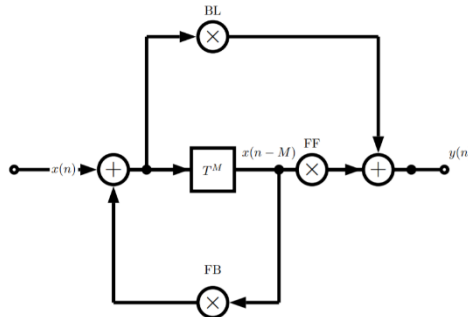


Рисунок 2 – Схема алгоритма работы устройства эффекта эхо

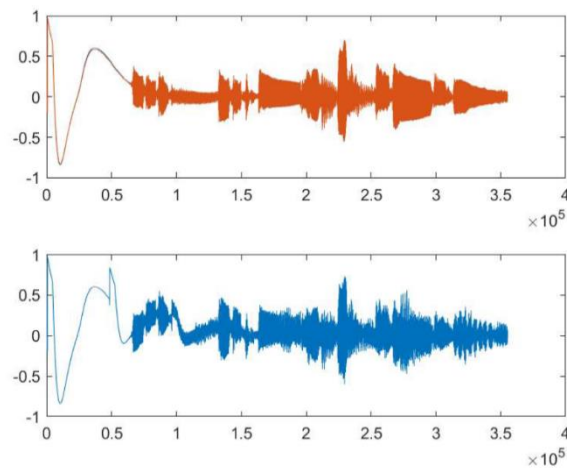


Рисунок 3 – Пример работы эффекта эхо во временной области

Суть эффекта overdrive – в ограничении сигнала по амплитуде. Функционально зависимость выходного сигнала от входного может быть представлена следующей формулой [1].

$$f(x) = \begin{cases} 2x & \text{for } 0 < x \leq 1/3 \\ \frac{3 - (2 - 3x)^2}{3} & \text{for } \frac{1}{3} \leq x \leq 2/3 \\ 1 & \text{for } \frac{2}{3} \leq x \leq 1 \end{cases}$$

Суть эффекта distortion также заключается в ограничении сигнала по амплитуде, но ключевое отличие в том, что overdrive обеспечивает «мягкое» ограничение. Функциональная зависимость выходного сигнала от входного с эффектом distortion представлен на формуле ниже. Ответ системы с эффектом distortion на синусоидальное воздействие представлен на рисунке 4. Зависимость выходного сигнала от входного эффекта перегруз (overdrive) представлена на рисунке 5.

$$f(x) = x \frac{x}{|x|} \left( 1 - e^{-\frac{ax^2}{|x|}} \right)$$

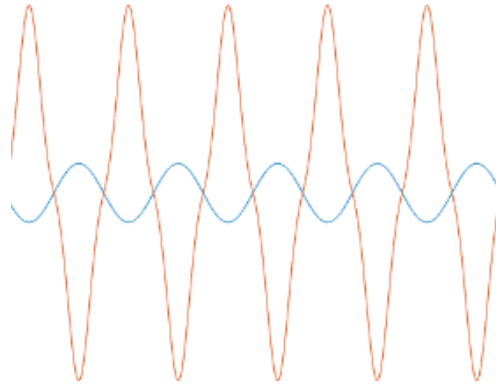


Рисунок 4 – Зависимость выходного сигнала от входного эффекта distortion

На рисунке можно заметить сдвиг фазы на  $180^\circ$ , что и подразумевается при использовании данного эффекта

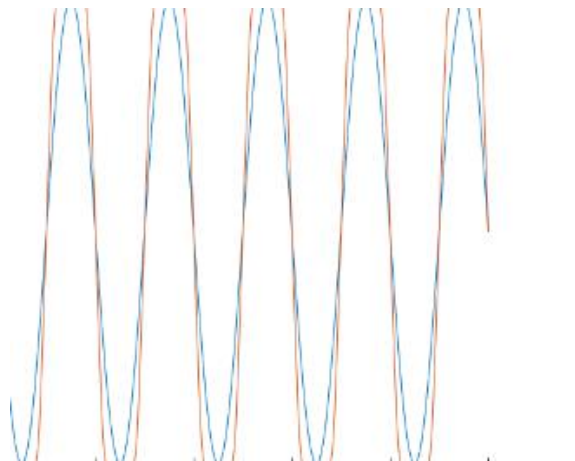


Рисунок 5 – Зависимость выходного сигнала от входного эффекта overdrive

Были промоделированы три гитарных эффекта: эхо, overdrive, distortion. Полученные в ходе исследования принципов работы гитарных эффектов графики полностью совпадают с описанными в методических материалах, что свидетельствует о возможности эмулирования любого гитарного эффекта, путём использования цифровой техники.

**Список использованных источников:**

1. Audio Effects: Theory, Implementation and Application Joshua D. Reiss, Andrew P. McPherson –CRC Press, 2015. – 355 P.

## МЕТОДЫ РЕАЛИЗАЦИИ ДВУМЕРНОГО ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ

*Шкет Н. Н.*

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Петровский Н.А. – к.т.н., доцент*

Двумерное вейвлет-преобразование используется в стандарте JPEG2000, который находит своё применение во множестве областей мультимедиа, включающих цифровое кино (digital cinema) [1]. В данной работе исследуются методы реализации двумерного вейвлет-преобразования с точки зрения производительности и латентности при выполнении, что критично для данного применения, а также с расчётом на реализацию на ПЛИС.

Вейвлет-преобразование представляет собой разложение сигнала на коэффициенты вейвлет-функций. Вейвлет-функция является конечной во времени и может быть использована для измерения частотно-временных изменений спектральных компонент. Вейвлет-преобразование также

эффективно применяется для выделения прерывистых непериодических изменений в сигнале путём представления на разных уровнях разложения [2].

Вейвлет-преобразование может быть представлено в виде эквивалентного банка фильтров состоящего из двух каналов, реализующих ФНЧ и ФВЧ. Схема одномерного вейвлет-преобразования представлена на рисунке 1.

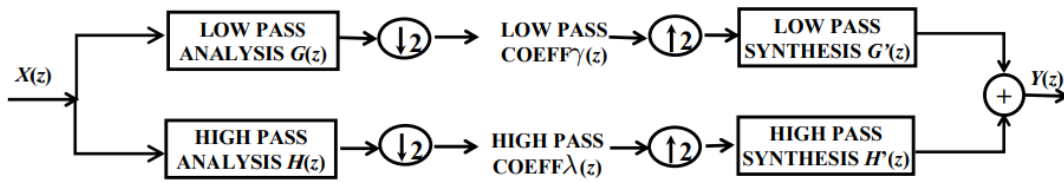


Рисунок 1 – Структура одномерного вейвлет-преобразования [3]

Для перехода от одномерного к двумерному вейвлет-преобразованию необходимо выполнить одномерное преобразование дважды – по строкам и по столбцам, порядок значения не имеет. Для сравнения количества операций, требуемых для разных структур, возьмём стандартное преобразование 9/7, которое используется при преобразовании с потерями в стандарте JPEG2000. Прямое преобразование представляет из себя симметричный ФНЧ 9 порядка и симметричный ФВЧ 7 порядка, обратное – симметричный ФНЧ 7 порядка и симметричный ФВЧ 9 порядка, соответственно.

Для сокращения количества операций используется лестничная структура. Рассматриваемое двумерное вейвлет-преобразование на основе лестничной схемы показано на рисунке 2.

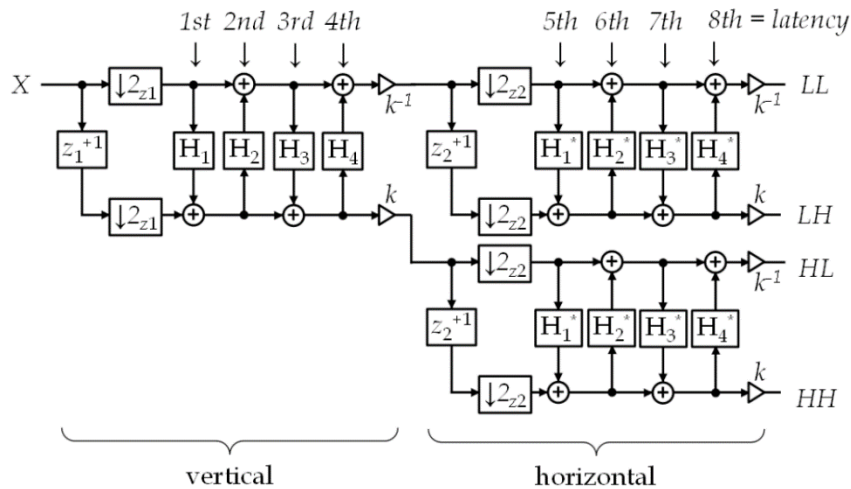


Рисунок 2 – Лестничная структура двумерного вейвлет-преобразования

Как следует из представленной на рисунке 2 схемы, лестничная структура требует 4 операции сложения с накоплением и две параллельные операции умножения на поправочный коэффициент для преобразования 9/7. Тем не менее, схемы использующие одномерное вейвлет-преобразование по строкам и по столбцам отдельно имеют серьёзное ограничение – большую латентность между этапами по строкам и по столбцам, а также память для промежуточного буфера размером с исходное изображение. При реализации преобразования на ПЛИС исходные данные поступают построчно и для уменьшения задержки при обработке логично не ожидать накопления изображения в буфере, а обрабатывать данные сразу при поступлении в двух измерениях.

Для устранения перечисленных ограничений предлагается использовать модифицированный разделимый и неразделимый подход в получении коэффициентов вейвлет-преобразования. Схемы потока данных для существующих лестничных структур представлены на рисунке 3.

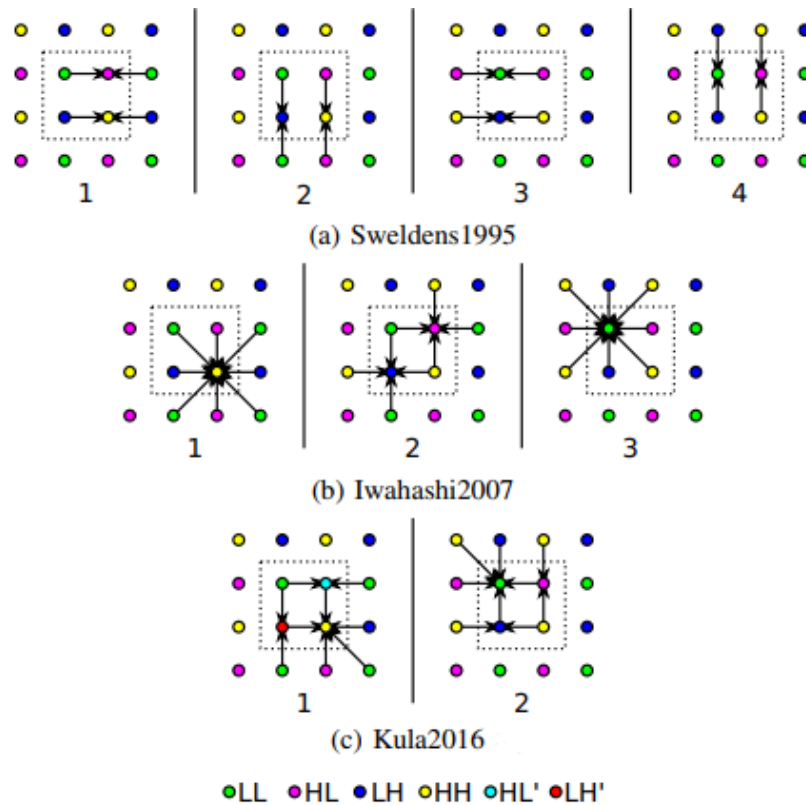


Рисунок 3 – Схемы потока данных рассматриваемых структур двумерного вейвлет-преобразования [4]

На рисунке 3 показаны ступени преобразования. Чтобы выполнить преобразование 9/7, нужно повторить эти ступени дважды. Как видно из схем потока данных, каждое преобразование использует будущие данные как по строкам, так и по столбцам. Следовательно, допуская, что данные поступают построчно, задержка на выполнение преобразования зависит от количества ступеней, использующих будущие данные. Таким образом, для изображения с разрешением  $N \times M$  точек, задержки на преобразовании 9/7 с помощью представленных на рисунке 3 схем, без учёта задержек самих вычислений, указаны в таблице 1.

Таблица 1 – Задержки на выполнение преобразования 9/7 для разных схем преобразования.

Преобразование	Sweldens1995	Iwahashi2007	Kula2016
Задержка	$4M+3$	$7M+3$	$3M+3$

**Список использованных источников:**

1. Applications for JPEG 2000 [Электронный ресурс]. – Режим доступа: <https://jpeg.org/jpeg2000/applications.html>
2. Mallat S., A wavelet tour of signal processing // Academic press, 1999. – P. 3-4.
3. Jin, Li. The Mathematics of JPEG2000 / Li Jin. – Modern Signal Processing, 2003. – №46. – С. 185-221.
4. Kula, M. New Non-separable Lifting Scheme for Images / M. Kula, D. Barina, P. Zemcik // 2016 IEEE International Conference on Signal and Image Processing (ICSIP), pp 292-295.



*Научное издание*

# **КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ**

**56-я научная конференция  
аспирантов, магистрантов и студентов**

Сборник тезисов докладов

(Минск, 21-24 апреля 2020 года)

В авторской редакции

Ответственный за выпуск *Е.В. Калабухов*

Компьютерная верстка: *А.А. Артамонов*