

ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ КОМПЬЮТЕРНЫХ НАУК И ТЕХНОЛОГИЙ

ИУС
МКМ 2019

СБОРНИК МАТЕРИАЛОВ
X МЕЖДУНАРОДНОЙ НАУЧНО-ТЕХНИЧЕСКОЙ КОНФЕРЕНЦИИ

**“ИНФОРМАТИКА,
УПРАВЛЯЮЩИЕ СИСТЕМЫ,
МАТЕМАТИЧЕСКОЕ
И КОМПЬЮТЕРНОЕ
МОДЕЛИРОВАНИЕ”**

(СТУДЕНЧЕСКАЯ СЕКЦИЯ)

В РАМКАХ V МЕЖДУНАРОДНОГО НАУЧНОГО ФОРУМА
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ

22-24 МАЯ 2019 Г.
ДОНЕЦК

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ**

**ГОУ ВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**ФАКУЛЬТЕТ
КОМПЬЮТЕРНЫХ НАУК И ТЕХНОЛОГИЙ**

**ИНФОРМАТИКА, УПРАВЛЯЮЩИЕ СИСТЕМЫ,
МАТЕМАТИЧЕСКОЕ И КОМПЬЮТЕРНОЕ
МОДЕЛИРОВАНИЕ
(ИУСМКМ-2019)
(Студенческая секция)**

Материалы студенческой секции X Международной
научно-технической конференции в рамках
V Международного Научного форума
Донецкой Народной Республики

22-24 мая 2019 г.

Научное издание

**ИНФОРМАТИКА, УПРАВЛЯЮЩИЕ
СИСТЕМЫ, МАТЕМАТИЧЕСКОЕ
И КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ
(ИУСМКМ-2019)**

Материалы студенческой секции X Международной
научно-технической конференции в рамках
V Международного Научного форума
Донецкой Народной Республики

22-24 мая 2019 г.

Web-сайт конференции: <http://www.iuskm.donntu.org>

УДК 004

Материалы студенческой секции X Международной научно-технической конференции «Информатика, управляющие системы, математическое и компьютерное моделирование» (ИУСМКМ - 2019). – Донецк: ДОННТУ, 2019. – 494 с.

Рецензенты: Карабчевский В.В., к.т.н., доц., зав.каф. КМД; Павлыш В.Н., д.т.н., проф., зав.каф. ПМ; Николаенко Д.В., к.т.н., доц., доц. каф. КИ; Зори С.А., д.т.н., доц., проф. каф. ПИ; Павлий В.А., к.т.н., доц. каф. КМД; Орлов Ю.К., к.т.н., доц., доц. каф. ИИСА; Копытова О.М., к.ф-м.н., доц., доц. каф. ИИСА; Мальчева Р.В., к.т.н., доц., проф. каф. КИ, зам.декана по науке ФКНТ ГОУВПО ДонНТУ; Светличная В.А., к.т.н., доц., доц. каф. АСУ; Бельков Д.В., к.т.н., доц., доц. каф. ПМ; Григорьев А.В., к.т.н., доц., проф. каф. ПИ.

Сборник подготовлен по результатам работы студенческой секции X Международной научно-технической конференции «Информатика, управляющие системы, математическое и компьютерное моделирование», проведенной в рамках V Международного Научного форума Донецкой Народной Республики.

Материалы, вошедшие в сборник, представлены научно-педагогическими сотрудниками, аспирантами, магистрантами и студентами высших учебных заведений из России, Беларуси, и ДНР.

Рекомендовано к публикации на заседании совета факультета КНТ ДонНТУ.
Протокол №5 от «21» июня 2019 г.

Адрес оргкомитета:
г. Донецк, ул. Кобозева, 17, Донецкий национальный технический университет, 4
учебный корпус, ФКНТ, кафедра КМД, ком. 4.41.
E-mail: iuskm_19@mail.ru

СОДЕРЖАНИЕ

	стр.
СЕКЦИЯ №1 ТЕОРЕТИЧЕСКАЯ И ПРИКЛАДНАЯ МАТЕМАТИКА.....	11
Алгоритм шифрования текстовых сообщений на основании графических файлов.	
<i>Анохина И.Ю., Лапшина Е.В.</i>	<i>12</i>
История развития прикладной математики.	
<i>Гончарова Е.Д., Добровольский Ю.Н.</i>	<i>17</i>
Применение информационных технологий для нахождения максимального компонента состава пород по пластам.	
<i>Прокопенко Е.В., Исаков А.Д.</i>	<i>20</i>
Теория алгоритмов как один из необходимых инструментов программиста.	
<i>Канатуш С.В., Ефименко К.Н.</i>	<i>25</i>
Задачи оптимизации расписания программ обработки деталей для станков ЧПУ.	
<i>Кононенко Д.Н., Лазебная Л.А., Бельков Д.В.</i>	<i>29</i>
Модель Бака-Снеппена для транспортного затора.	
<i>Пауков Д.В., Зензеров В.И., Бельков Д.В.</i>	<i>34</i>
Великая теорема Ферма и ее влияние на развитие современной науки.	
<i>Потребя Е.Ю., Ефименко К.Н.</i>	<i>38</i>
Шифрование символьной информации на машине Тьюринга.	
<i>Шиян А.С., Добровольский Ю.Н.</i>	<i>43</i>
Выбор и обоснование программно-математического аппарата для создания геометрической модели породного отвала.	
<i>Прокопенко Е.В., Юрко Ю.С.</i>	<i>47</i>
СЕКЦИЯ №2 ПРОГРАММНАЯ ИНЖЕНЕРИЯ.....	51
Тестирование мобильного приложения на группе виртуальных устройств.	
<i>Безсонов И.В., Гудаев О.А.</i>	<i>52</i>
Обзор существующих протоколов для программной реализации голосового мессенджера.	
<i>Вязмин В.И., Чернышова А.В.</i>	<i>55</i>
Применение цифровых водяных знаков с целью защиты авторских прав на графические изображения.	
<i>Шевляков А.И., Ефремченко И.А., Губенко Н.Е.</i>	<i>60</i>
Музыкальный плеер с функцией синхронизации ритма музыки с ритмом шага человека.	
<i>Комаричев Р. Е., Чернышова А. В.</i>	<i>64</i>
Проектирование и разработка приложения для аудио стеганографии.	
<i>Толбатова А.С., Чернышова А.В.</i>	<i>70</i>
Возможности поиска документов об образовании в программном комплексе «Государственный реестр сведений о документах об образовании».	
<i>Рубан О.И., Кравченко Т.М.</i>	<i>76</i>
Программное обеспечение учёта и журналирования для спортивных комплексов.	
<i>Полуденный Н.И.</i>	<i>80</i>
СЕКЦИЯ №3 СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ.....	86
Исследование технологий создания интеллектуальных надстроек над визуальными редакторами типа Visio.	
<i>Ржевский К.В., Григорьев А.В.</i>	<i>87</i>
Применение имитационного моделирования для усовершенствования управления транспортными потоками города и безопасности на пешеходных переходах.	
<i>Розанова С.А., Косенко Ю.А., Незамова Л.В.</i>	<i>91</i>

3D-принтер, анализ современного состояния, перспектив развития и практического применения.	
<i>Харламов И.А., Григорьев А.В.</i>	96
Механизмы движения и физические зависимости в современных САПР.	
<i>Чернышов Д.Н., Григорьев А.В.</i>	101
Исследование API САПР CATIA V5 и перспективы развития создания приложений в ней.	
<i>Бондаренко Е.С., Григорьев А.В.</i>	105
Проблематика моделирования краш-тестов для моделей мотоциклов.	
<i>Шевелёв И.А., Брюхова Д.В., Григорьев А.В.</i>	110
СЕКЦИЯ №4 ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ.	114
Двухуровневый генетический алгоритм оптимизации грузоперевозок.	
<i>Бражник А.В., Секирин А.И., Валуева О.С.</i>	115
Методы анализа и оптимизации работы кухни в условиях непрерывного производства.	
<i>Озеров А.О., Хмелевой С.В.</i>	122
Определения количества транспортных средств с помощью имитационной модели и методов принятия решений.	
<i>Шиленко А.В., Светличная В.А.</i>	127
Использование имитационного моделирования при календарном планировании.	
<i>Стрельников Е.А., Светличная В.А.</i>	132
Разработка функциональной модели Web-ориентированной системы бронирования различных туристических услуг.	
<i>Ясницкий М.В., Андриевская Н.К.</i>	137
Автоматизированная система контроля и управления доступом на основе анализа биометрических параметров человека.	
<i>Антипов Р.С., Мартыненко Т.В.</i>	143
Инструментальные средства для компьютерной обработки изображений.	
<i>Кисловская Я. О., Землянская С. Ю.</i>	148
Разработка базы данных алгоритмов.	
<i>Павлов М.В., Харитонов Ю.Е.</i>	154
Плагиат. Нарушение и защита авторских прав.	
<i>Ищук Л.А.</i>	158
Подсистема мониторинга и контроля усталости водителя.	
<i>Кожухов А.С., Васяева Т.А.</i>	161
Web-ориентированная система планирования и учета научно-технических мероприятий в ДонНТУ.	
<i>Чайка В.А., Землянская С.Ю.</i>	166
Web-ориентированная система мониторинга состояний заказов при приобретении товаров в интернет-магазине.	
<i>Калюга Д.В., Поляков А.И., Теплова О.В.</i>	171
Современные тенденции в технологии машинного обучения.	
<i>Егорова М.С., Мартыненко Т.В.</i>	175
Автоматизированная система учета заказов клиентов/посетителей ресторана.	
<i>Головин П.В., Поляков А.И., Пряхин В.В.</i>	181
Разработка мобильного приложения StepsTracker (шагомер) в MIT App Inventor.	
<i>Харитонов Ю.Е., Денисенко Е.А.</i>	185
Система распознавания лжи по мимике человека на основе видео.	
<i>Лютый В.А., Секирин А.И.</i>	189

стр.

Автоматизированный подход к выбору радиологических карт для лучевой терапии. <i>Ежело С.Е., Васяева Т.А. ..</i>	195
Использование имитационного моделирования для формализации задач принятия решений. <i>Барабанищikov О.Е., Савкова Е.О.</i>	200
Управление транспортными потоками с использованием адаптивных алгоритмов. <i>Довидович В.С.</i>	205
Обнаружение угроз безопасности в системе контроля и управления доступом при помощи деревьев решений. <i>Юрьев Н. Н., Васяева Т. А.</i>	210
Разработка модели интеллектуальной системы формирования цен в интернет-магазине. <i>Лисничук В.Е., Землянская С. Ю.</i>	214
Автоматизированная система сегментации легких с использованием нейронной сети. <i>Пашков А.С., Васяева Т.А.</i>	219
СЕКЦИЯ №5 МАТЕМАТИЧЕСКОЕ И КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ.....	222
Разработка имитационной модели колл-центра в среде AnyLogic. <i>Нелюбина Ю.А., Харитонов Ю.Е.</i>	223
СЕКЦИЯ №6 ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ СИСТЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА.....	228
Анализ принципов работы лингвистического процессора в обработке естественных языков. <i>Михневич П.Д.....</i>	229
Разработка модуля системы распознавания рукописных символов. <i>Войдолович И.В., Бычкова Е.В.</i>	231
Обзор алгоритмов построения кратчайшего пути. <i>Горлов Е.Д., Березовская Е.А., Миненко А.С., Едемская Е.Н.....</i>	234
Программная реализация алгоритма «FOREL 2» в файловых системах с FAT таблицами. <i>Добренко Д.С., Ольшевский А.И.....</i>	238
Анализ модели нейрона Бонхоффера – Ван дер Поля. <i>Дробот К.С., Едемская Е.Н., Бычкова Е.В., Бельков Д.В.</i>	242
Исследование и разработка алгоритма для рекомендательной системы. <i>Орлов Ю.К., Жулидов А.Ю.</i>	246
Система управления взаимоотношениями с клиентами. <i>Калужский Г.А., Ольшевский А.И.</i>	249
О влиянии переброски дуг на деградацию поведения конечного автомата. <i>Ковалёв Д.В., Копытова О. М.</i>	252
Исследование и разработка методов генерации трёхмерных изображений при помощи нейронных сетей. <i>Кошман А.Д., Ольшевский А.И.....</i>	256
Сегментация изображения на основе метода оптимизация роя частиц. <i>Купоросов А.А.</i>	261
Архитектура интернет вещей на базе технологии блокчейна. <i>Матвеев М.О., Гудаев О.А.</i>	266
Архитектура программного комплекса резервного копирования данных. <i>Ольшевский А.И., Нестеренко В.С.</i>	270

Исследования разработки динамического языка жестов человека для управления мультимедийным контентом.	
<i>Гетманова А.Д., Копытова О.М.</i>	275
Моделирование и анализ мимических проявлений эмоций на лице человека.	
<i>Пойда А.А.</i>	279
Создание Web-сайта для магазина строительных материалов.	
<i>Титов К.В., Радевич Е.В.</i>	283
Методика распознавания печатного текста с выделением графического материала.	
<i>Шумский А.А., Кравец Т.Н.</i>	286
Проектирование и разработка информационной системы предприятия оптовой торговли с использованием программы «1С:Предприятие» версии 8.2.	
<i>Ярмак М.А., Семенова А.П.</i>	290
Разработка экспертной системы для проверки веб-сайтов в качестве SEO технологий.	
<i>Деньзуб Ю.О., Гудаев О.А.</i>	294
Использование технологий искусственного интеллекта при игре в покер.	
<i>Котлубеев А.Ю., Головченко А.А., Бочаров Д.М.</i>	296
Алгоритм построения карты для робота на основе данных ультразвуковых датчиков.	
<i>Егурнов Н.О., Миненко А.С.</i>	300
Разработка онтологического чат-бота на базе глубинного обучения нейронных сетей.	
<i>Оверченко Я.Ю., Копытова О.М.</i>	304
Разработка обучающего пакета программ по курсу «Операционные системы».	
<i>Ищенко Р.Д., Ольшевский А.И.</i>	307
Определение требований к разрабатываемому интернет-магазину.	
<i>Проничев А.А., Безрукавый И.В., Семенова А.П., Бочаров Д.М.</i>	311
СЕКЦИЯ №7 КОМПЬЮТЕРНАЯ ГРАФИКА И ДИЗАЙН	317
Разработка системы «Увлекательная география» с использованием моделей дополненной реальности в межплатформенной среде Unity.	
<i>М.Б. Бобелюк, Н.Е. Губенко.</i>	318
Analysis of models and methods for studying multimedia products customer satisfaction.	
<i>A.O. Kandaurova, N.Ye. Gubenko, Ye.N. Kushnirenko.</i>	322
Методы анализа удовлетворенности потребителей рекламных услуг.	
<i>Кандаурова А.О., Губенко Н.Е.</i>	327
Анализ дизайн индустрии и бизнес-процессов в дизайн-студиях.	
<i>Тилинина Н.Ю., Губенко Н.Е.</i>	332
Разработка игры «Логический баскетбол».	
<i>Лащенко В.Э., Киселева О.В.</i>	335
Повышение защищенности конфиденциальной информации на основе комбинации методов жаргонов.	
<i>Сидорчук В.И., Губенко Н.Е.</i>	338
Влияние SMM-дизайна на продвижение и восприятие бренда.	
<i>Шепеленко Н.С., Руденко М.П.</i>	343
Разработка уровня игры «Занимательная математика» средствами Construct 2.	
<i>Беспалова Е.С., Киселева О.В.</i>	350
Разработка сцен игры «Занимательная грамматика» средствами Unity 3D.	
<i>Демьяненко В.А., Киселева О.В.</i>	354
Геометрическое и динамическое моделирование сложного промышленного 3D-ландшафта в среде Cinema 4D.	
<i>Верховский А.В., Пчелкин В.Н.</i>	360

Использование игровых методов в обучающих системах по развитию навыков программирования.	
<i>Богущая В.Б., Губенко Н.Е.</i>	364
Создание информационного сайта с возможностью составления индивидуальной программы тренировок.	
<i>Катков М.В., Павлий В.А.</i>	369
Электронные библиотеки и потребности пользователей. Исследование рекомендаций по дизайну электронных библиотек и разработка требований на их основе.	
<i>Донцова Е.А.</i>	374
СЕКЦИЯ №8 КОМПЬЮТЕРНАЯ ИНЖЕНЕРИЯ	379
Анализ тестируемости константных неисправностей на базе т-модели объекта диагностики.	
<i>Лактионов А. А., Зинченко Ю. Е., Зинченко Т. А.</i>	380
Аппаратная обработка и распознавание звуковых сигналов на базе FPGA.	
<i>Азаров А. Б., Константинов В. С., Зинченко Ю. Е., Зинченко Т. А.</i>	386
Исследование данных жёсткого диска с использованием инструментов UEFI.	
<i>Чередникова О.Ю., Якубов Я.А.</i>	391
Методы разграничения прав доступа к удаленным ресурсам.	
<i>Чередникова О. Ю., Зимцев П. П.</i>	395
Взаимодействие различных SCADA-систем в цифровых промышленных сетях.	
<i>Чередникова О.Ю., Коротач А.С.</i>	400
Разработка утилиты анализа схемы разметки жесткого диска.	
<i>Чередникова О.Ю., Красников Р.С.</i>	404
Структура хранения информации о составе каталога в файловой системе NTFS.	
<i>Чередникова О.Ю., Уткин П.С.</i>	408
Виды нейронных сетей и способы их обучения.	
<i>Щёголев В. А., Чередникова О. Ю.</i>	414
Применение ЧПУ для изготовления заказных печатных плат.	
<i>Захарченко К. С., Достлев Ю. С.</i>	418
Особенности применения датчика температуры и влажности DHT11.	
<i>Евтушенко О. В., Кривошеев С. В.</i>	422
Программная модель температурного датчика DS18B20.	
<i>Симак А. С., Кривошеев С. В.</i>	425
СЕКЦИЯ №9 СИСТЕМНЫЙ АНАЛИЗ И УПРАВЛЕНИЕ. ПРИКЛАДНАЯ ИНФОРМАТИКА	431
Разработка структуры виртуальной лаборатории по вычислительной геометрии.	
<i>Бурзак К.А., Едемская Е.Н.</i>	432
Анализ внедрения системы электронного документооборота (далее СЭД) университета с использованием методологии функционального моделирования IDEF0.	
<i>Бутенко Д.А., Радевич Е.В.</i>	437
Программная реализация системы нечетких правил для автоматизации отдельных узлов металлургического производства.	
<i>Гуляев В.А., Орлов Ю.К.</i>	443
Краткий обзор платформы arduino uno и ультразвукового датчика измерения расстояния hc-sr04.	
<i>Егурнов Н.О., Миненко А.С.</i>	449
Исследование и разработка системы диагностики технических объектов.	
<i>Орлов Ю.К., Свиридов Д.В.</i>	452

Электронный документооборот.	
<i>Корнеева Е.А., Папин О.В., Едемская Е.Н., Сорокин Р.А.</i>	456
Создание программы для робота и реализация его функционирования.	
<i>Перепелица Р.А., Давиденко А.Н., Куликов Д.Р., Елисеев В.И.</i>	460
Создание программы для участия в соревновании «Кегельринг».	
<i>Перепелица Р.А., Давиденко А.Н., Куликов Д.Р., Елисеев В.И.</i>	463
Прогноз и оценка риска возникновения чрезвычайной ситуации на угледобывающих предприятиях.	
<i>Резникова Г.А., Орлов Ю.К.</i>	466
Прогнозирование объёмов продаж в Excel.	
<i>Хоменко В.В., Орлов Ю.К.</i>	470
Разработка модели работы системы потребления топливно-энергетических ресурсов.	
<i>Орлов Ю.К., Чайка А.Р.</i>	475
Моделирование процесса кристаллизации металла в электрошлаковом переплаве.	
<i>Миненко А.С., Шихов И.В.</i>	479
Разработка системы планирования закупок товаров медицинского назначения на коммерческом предприятии.	
<i>Удовика Ю.Д., Тарасова И.А.</i>	482
Методы прогнозирования и анализ фондового рынка.	
<i>Колесников В.Ю., Ванжа Т.В.</i>	485
Разработка системы построения дерева синтаксического подчинения для анализа русскоязычных текстов.	
<i>Лёвкин А.В., Тарасова И.А.</i>	489

СЕКЦИЯ 1

ТЕОРЕТИЧЕСКАЯ И
ПРИКЛАДНАЯ МАТЕМАТИКА

ИУС
МКМ 2019

УДК 681.3.06

АЛГОРИТМ ШИФРОВАНИЯ ТЕКСТОВЫХ СООБЩЕНИЙ НА ОСНОВАНИИ ГРАФИЧЕСКИХ ФАЙЛОВ

Анохина И.Ю., Лапшина Е.В.

Донецкий национальный технический университет
кафедра прикладной математики
E-mail: ingatula@mail.ru

Аннотация:

Анохина И.Ю., Лапшина Е.В. Алгоритм шифрования текстовых сообщений на основании графических файлов. Рассмотрены вопросы классификации криптографических алгоритмов, задачи симметричного шифрования. Описан алгоритм шифрования текстовых сообщений на основании передаваемых графических файлов.

Annotation:

Anokhina I.Yu., Lapshina E.V. Text message encryption algorithm based on graphic files. The classification of cryptographic algorithms, the problem of symmetric encryption. An algorithm for encrypting text messages based on transferred graphic files is described.

Введение. Для защиты информации используют криптографические методы. Криптография – наука о методах, алгоритмах, программных средствах, используемых для преобразования информации в целях ее сокрытия (искажения), предотвращения ее видоизменения и несанкционированного использования.

Целью криптографической системы является шифрование передаваемого сообщения. **Шифрование** – это преобразование сообщения в код, при этом должно быть обеспечено однозначное соответствие между символами текста и кода. **Дешифрование** – обратный процесс преобразования кода в сообщение с помощью ключа. **Ключ** - информация, необходимая для дешифрования. Степень сложности системы для дешифрования без известного ключа, называется надежностью [1].

Шифрование сообщений используется с тех незапамятных времен, когда появилась первая секретная информация, доступ к которой ограничивался определенным кругом лиц, поэтому проблема разработки все новых алгоритмов шифрования/дешифрования остается актуальной и по сей день.

В настоящее время криптография переживает невиданный подъем, что обуславливается двумя факторами. Первый - развитие вычислительной техники привело к необходимости защиты именно компьютерной информации, второй фактор обусловлен тем, что если ранее, как правило, к шифрованию прибегали государственные структуры, то сейчас это используют обычные люди и организации.

С учетом того количества конфиденциальной информации, которая пересылается в настоящее время, ясно, что проблема защиты конфиденциальной информации становится все острее. Об этом свидетельствует и большое количество разработанных алгоритмов, программ, онлайн сервисов для шифрования/дешифрования информации.

Криптоалгоритмы можно разделить на три категории:

- бесключевые, не использующие ключей в процессе шифрования;
- одноключевые с одним секретным ключом;
- двухключевые с секретным и открытым ключом.

Обобщенно схема криптосистемы показана на рис.1. Исходный текст M (message) зашифровывается с помощью преобразования E (encryption) с использованием ключа K (key). В результате получают C (cipher text) – зашифрованное сообщение. Таким образом, предполагается наличие трех компонентов: данных, ключа и криптографического преобразования.

Законный получатель с помощью обратного преобразования $D(E^{-1})$ получает исходное сообщение.

$$C = E_K(M); M' = D_K(C).$$

Такой вид криптосистемы называется симметричной [2]. Стойкость криптосистемы зависит от сложности алгоритмов преобразования и объема ключевого пространства. Ясно, что методом перебора по всему ключевому пространству можно дешифровать практически любое сообщение, но у противника может не хватить времени и ресурсов для выполнения задачи [3].

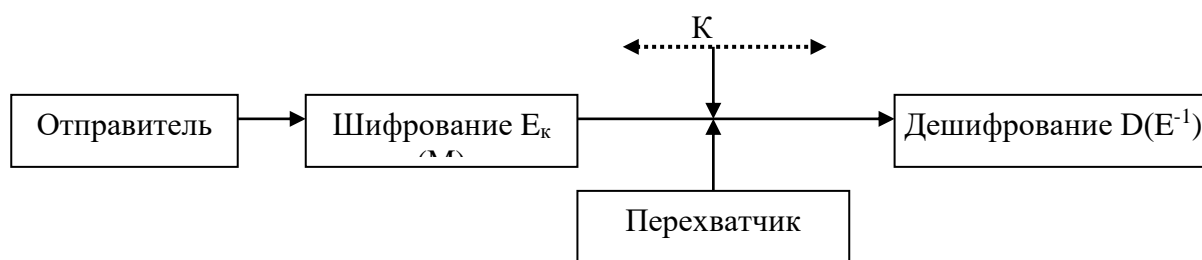


Рис. 1. Схема симметричной криптосистемы с одним ключом

Постановка проблемы. Одним из недостатков стандартных методов шифрования является то, что закодированное сообщение привлекает внимание, предполагая наличие ценной информации. Поэтому целесообразно скрывать наличие такой информации.

Одним из вариантов может быть шифрование сообщения в изображении, которое можно переслать в виде обычной картинка- поздравления.

Была поставлена задача разработать алгоритм шифрования текстовых сообщений в файлы формата bmp с возможностью извлечения сообщения без искажения.

Формат bmp – формат растровых изображений, в которых цвет задается в цветовой модели RGB (red – красный, green – зеленый, blue- синий). Значения каналов могут принимать значения от 0 до 255.

При такой постановке задачи в роли ключа выступает непосредственно файл bmp. Для усложнения несанкционированного дешифрования кроме картинки-ключа использовались различные алгоритмы выбора траектории кодирования.

Исследования. Характеристикой алгоритма шифрования является его криптографическая стойкость. Алгоритм считается криптографически стойким, если не существует каких-либо методов его вскрытия, кроме перебора всех возможных вариантов, а размер ключа достаточно велик, чтобы стало практически невозможно его использование. Нами в качестве ключа выбирается графический файл, следовательно, при попытке несанкционированного дешифрования возникает сразу несколько задач:

1. понять, какое именно изображение используется в качестве ключа;
2. определить алгоритм выбора траектории кодирования, т.е. какие именно координаты изображения были использованы для кодирования символов.

Разработанный алгоритм шифрования состоит из нескольких этапов:

1. загрузка изображения, используемого в качестве ключа;
2. анализ сообщения, которое должно быть зашифровано;

3. в зависимости от начальной буквы сообщения (гласная, согласная, «не текст») выбирается траектория кодирования;
4. каждому символу ставится в соответствие код;
5. кодируется пересылаемое сообщение.

Большинство современных криптоалгоритмов базируются на принципе Кирхгофа: секретность шифра обеспечивается секретностью ключа, а не секретностью алгоритма шифрования. В данном алгоритме секретность шифра обеспечивается не только ключом, но и секретностью алгоритма.

На рис. 2 показана блок-схема, отображающая процесс шифрования.

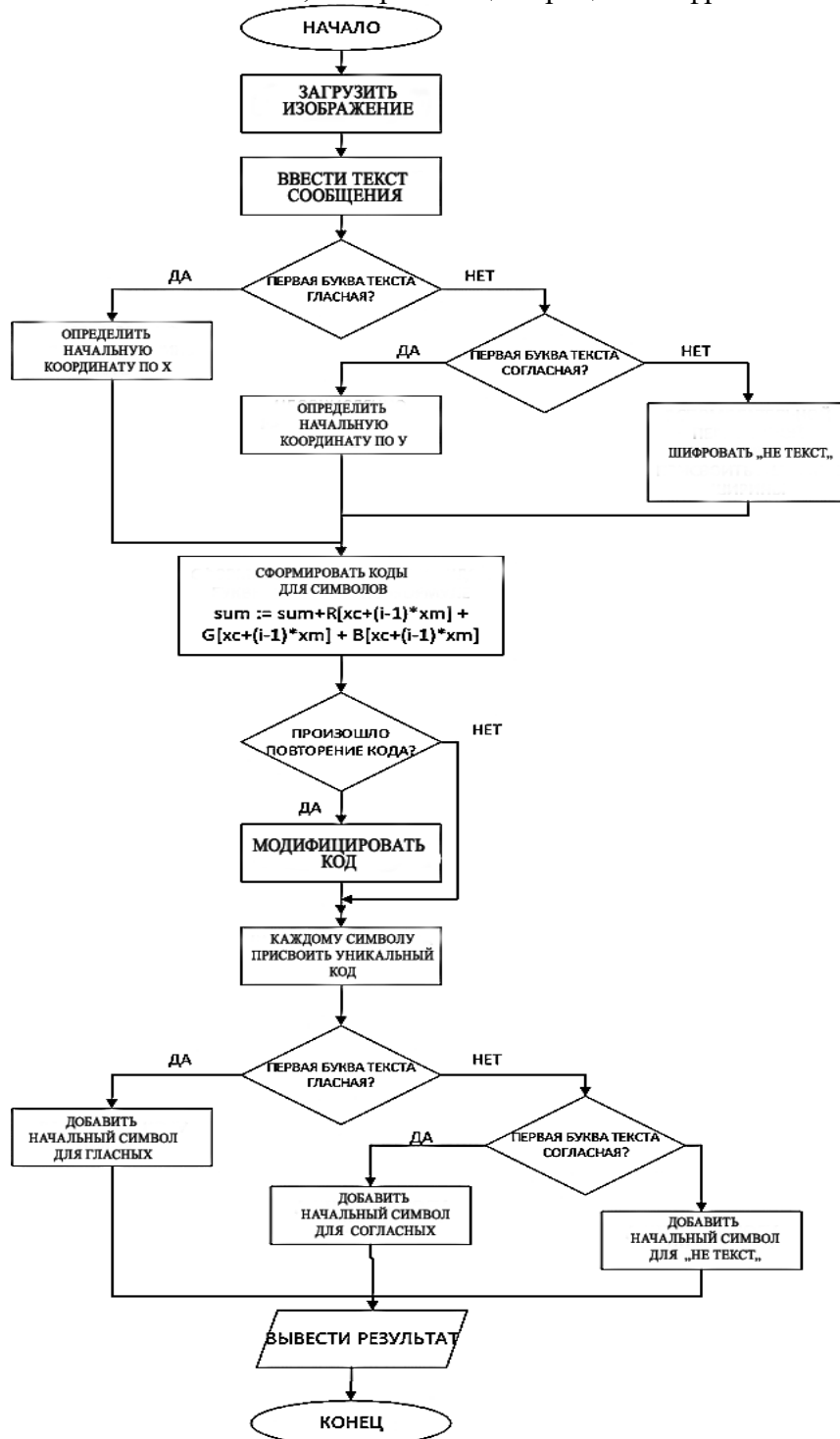


Рис. 2. Укрупненная блок-схема алгоритма

После выбора изображения определяются его размеры и формируются массивы, хранящие интенсивность базовых цветов в точках с координатами (X,Y) изображения. Они обозначались в программе как R, G и B (R – красный, G – зеленый, B – синий). При выполнении этих операций был использован тип TColor и его свойство Color, предназначенное для управления цветом объекта [4]. Каждая буква кодируется как сумма значений элементов массивов интенсивности цветов. При повторении значений сумм различных точек, она модифицируется.

Траектория выбора координат (X,Y) определяется в зависимости от первой буквы сообщения (гласная/согласная).

Для решения поставленной задачи разработан алгоритм и реализована программа в среде Delphi 7.

Для работы использованы три формы:

1. стартовая форма, позволяющая сделать выбор, будет ли проведено шифрование или дешифрование информации;
2. вторая форма использовалась для формирования массива кодов, шифрования, проверки на наличие ошибок;
3. третья форма использована для дешифрования информации.

На рис.3 показана вторая форма программы. Нами приведены результаты шифрования двух сообщений, первое начиналось с согласной, второе с гласной буквы. Результаты кодирования различны. Зашифрованные сообщения состоят из трехзначных цифр, соответствующих одной букве, разделенных пробелами. Первая цифра сообщения передает значение гласная/согласная для последующего дешифрования. Используя пункт меню «Проверка» имеется возможность осуществить дешифрование сообщения перед отправкой, чтобы убедиться в его правильности.

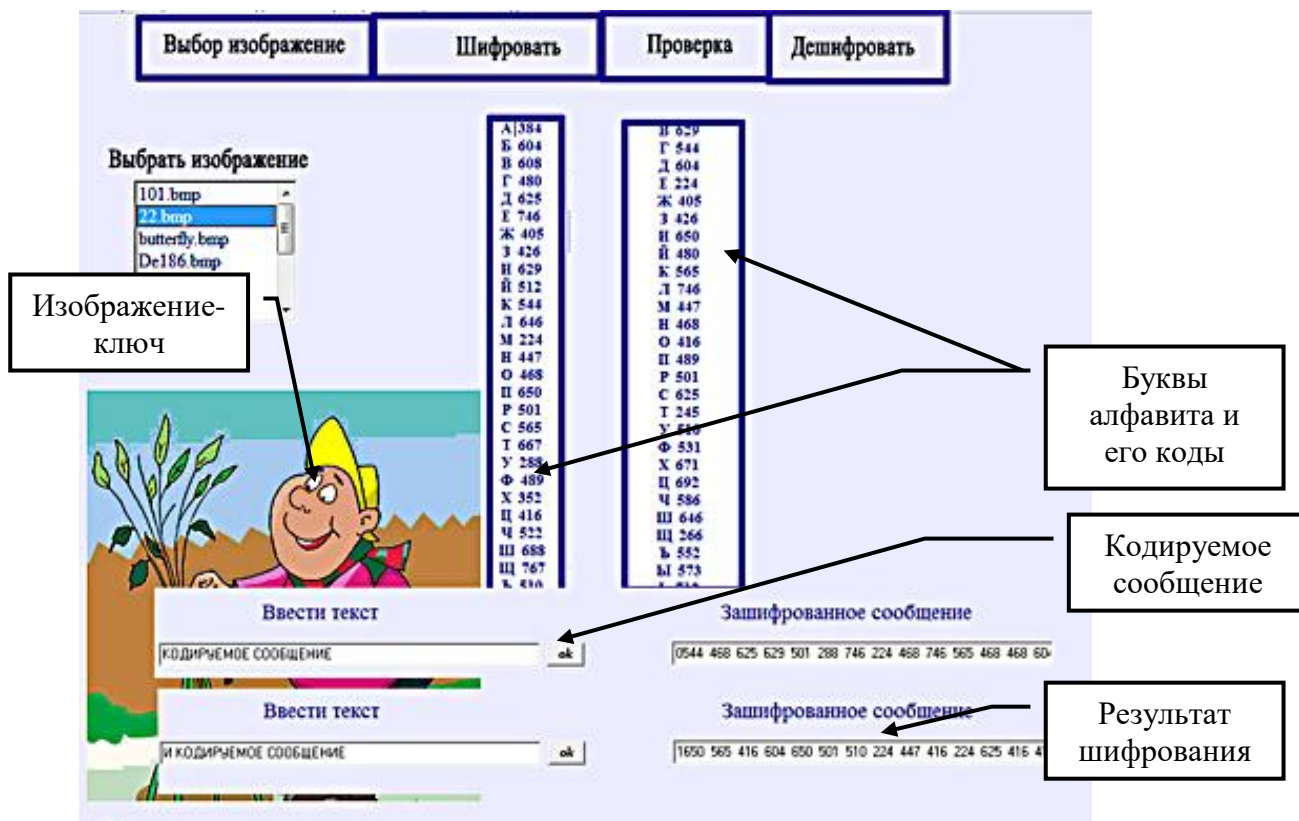


Рис. 3. Шифрование сообщения

Описанный алгоритм относится к категории алгоритмов симметричного шифрования, при таком виде алгоритма для операций шифрования/дешифрования используется один и тот же ключ. В процессе дешифрования считывается изображение - ключ, файл проверяется на идентичность, затем осуществляется обратная операция, по известным кодам формируется текстовое сообщение.

В таблице 1 показаны результаты шифрования сообщений на основании двух графических файлов.

Таблица 1 – Шифрование сообщений

Текст сообщения	Коды
Шифруем сообщение	0688 629 489 501 288 746 224 565 468 468 604 767 746 447 629 746
А мы все еще шифруем сообщение	1384 447 573 629 625 224 224 266 224 646 650 531 501 510 224 447 625 416 416 608 266 224 468 650 224
Шифруем сообщение	0688 629 489 501 288 746 224 565 468 468 604 767 746 447 629 746
А мы все еще шифруем сообщение	1337 353 366 389 420 383 383 933 383 360 390 441 870 891 383 353 420 411 411 765 933 383 849 390 383

Выводы

К преимуществам алгоритма следует отнести его достаточно высокую криптографическую стойкость, составляющими которой являются:

- неизвестное изображение-ключ;
- неизвестная траектория выбора точек, цвет которых будет раскладываться на составляющие;
- неизвестные варианты модификации траектории в зависимости от текста сообщения;
- возможность шифрования без доступа к Интернету, если у отправителя и получателя имеются изображения-ключи.

Программа проверена на ряде сообщений, сбоев обнаружено не было.

К недостаткам следует отнести то, что на данный момент таким способом можно зашифровывать лишь небольшие сообщения, работа по усовершенствованию алгоритма ведется.

Литература

1. Шаньгин В.Ф. Информационная безопасность компьютерных систем и сетей - М.: ИД ФОРУМ: ИНФРА-М, 2012. - 416 с.: ил.
2. Основные понятия криптографической защиты информации [электронный ресурс] // Режим доступа: URL: <http://ypn.ru/187/introducing-to-cryptographic-information-protection>. - Your Private Network. Лаборатория сетевой безопасности.
3. Панасенко С. П. Алгоритмы шифрования. Специальный справочник. — СПб.: БХВ-Петербург, 2009. — 576 с.: ил.
4. Архангельский А. Я. Программирование в Delphi 7. –М.:ООО «Бином-Пресс», 2003г.-1152.: ил.

УДК 519.6 (076)

ИСТОРИЯ РАЗВИТИЯ ПРИКЛАДНОЙ МАТЕМАТИКИ

Гончарова Е.Д.¹, Добровольский Ю.Н.²

Донецкий национальный технический университет
студентка гр. ИС-17а¹, кафедра прикладной математики²

E-mail: ielizadm@gmail.com

Аннотация:

Гончарова Е.Д., Добровольский Ю.Н. История развития прикладной математики. Рассмотрена краткая история развития прикладной математики, как самостоятельного раздела высшей математики для инженеров. Показана роль прикладной математики в решении ряда актуальных технических задач.

Annotation:

Goncharova E.D., Dobrovolsky Yu.N. History of applied mathematics. A brief history of the development of applied mathematics as an independent section of higher mathematics for engineers is considered. The role of applied mathematics in solving a number of relevant technical problems is shown.

Введение

Прикладная математика – область математики, рассматривающая применение математических методов, алгоритмов в других областях науки и практики. Примерами такого применения являются: численные методы, математическая физика, линейное программирование, оптимизация и исследование операций и т.д.

Возможность применения математики для решения прикладных задач является ее сущностью. Ведь само возникновение математики (алгебра, геометрия) несколько тысяч лет назад было обусловлено хозяйственными потребностями того времени. Например, по мнению историков, соотношение между сторонами прямоугольного треугольника было известно еще в Междуречье примерно за 1800 лет до н. э. Все последующее время развитие математики было тесно связано с решением технических проблем.

Цель данной работы – краткое изложение основных фактов, событий и идей в ходе многовековой истории развития математики в целом и одного из её важнейших направлений – «прикладной» (вычислительной) математики, зарождения и развития вычислительной техники и программирования.

Роль прикладной математики в развитии цивилизации

Существенный прогресс в развитии математики начался в эпоху Возрождения (XVII в. и последующие годы). Работы математиков этого периода охватывали много областей – новых и старых. Они обогатили оригинальными результатами классические разделы, пролили новый свет на прежние области и создавали даже совершенно новые области математических исследований. Примером первого рода могут служить труды Ферма. Новым творением была математическая теория вероятностей. Движущей силой в этом расцвете творческой науки, было не только то, что новой техникой можно было легко пользоваться. Многие крупные мыслители искали большего – «общего метода», который иной раз понимали в ограниченном смысле, как метод математики, иной раз понимали шире – как метод познания природы и создания новых изобретений. Это было причиной того, что в рассматриваемую эпоху все выдающиеся философы были математиками и все выдающиеся математики были философами [1]. В поисках новых изобретений они иногда непосредственно приходили к математическим открытиям. Выдающимися математиками

того времени были: Галилей, Декарт, Кеплер, Паскаль, Гюйгенс, Ньютон, Лейбниц, братья Бернулли, Эйлер, Лагранж, Лаплас, Гаусс.

Последующие успехи естествознания в конце XIX – начале XX веков сыграли решающую роль в подготовке научно-технической революции XX века. Революционный сдвиг произошёл в технике, в первую очередь под влиянием применения электричества в промышленности и на транспорте. Было изобретено радио, родилась авиация. Открытие электрона, радия, превращения химических элементов, создание теории относительности и квантовой теории ознаменовали прорыв науки в область микромира и больших скоростей. Человечество убедилось в колоссальных преобразующих возможностях науки и её практического применения. В начале XX века появились специальные курсы высшей математики для инженеров. Именно в эти годы началось формирование вычислительной и прикладной математики в качестве самостоятельных разделов математики [2].

В начале XX в. (в 1906 г.), выдающийся русский математик, механик и инженер-кораблестроитель академик А. Н. Крылов, обеспокоенный тем, что в современных курсах математического анализа доказываются существование решения какой-либо проблемы и теоретическая возможность получения его с любой степенью точности, но при этом не уделяется внимания получению такого решения с точностью, необходимой для практических целей, решил исправить это положение. Им был составлен курс о приближенных вычислениях, приемах и способах: вычисление корней численных уравнений, и определенных интегралов, пользование тригонометрическими рядами и приближенное решение дифференциальных уравнений.

Книга А.Н. Крылова «Лекции о приближенных вычислениях» была первым в мировой литературе курсом такого направления и послужила образцом для последующих, вышедших после нее курсов других авторов. Этот классический курс выдержал пять изданий (три из них при жизни автора). В главе IV описывается также общая теория и конструкция механических приборов для вычисления определенных интегралов. Это механические приборы, в которых исходные данные и результаты представляются не в цифровой форме, а в виде линейных или угловых перемещений частей этих приборов [3].

В докладе «Прикладная математика и техника», который А.Н. Крылов сделал на чрезвычайной сессии АН СССР 21 июня 1931 г., подчеркивалось, что «инженер должен владеть общими математическими методами, приложенными к решению множества задач, тогда только он сможет решать действительно новые вопросы по своей специальности...

Во всяком техническом деле важен не тот логический процесс, который привел к какому-либо заключению или результату, а важно самое заключение или самый результат и притом выраженный "числом и мерою". Поэтому все, что математика дает в смысле составления уравнений, их решения, и притом доведенного до конца, упрощения вычислений, применения приближенных методов решения математических вопросов, – все это техника рано или поздно использует и применит часто в вопросе, казалось бы, ничего общего не имеющем с тем, для решения которого тот или иной метод был первоначально развит...» [4].

Таким образом, к началу 40-х годов были разработаны математические методы решения ряда актуальных технических задач. Однако на этом пути имелись серьезные препятствия. Основная проблема – это примитивность вычислительной техники, которая не позволяла быстро и точно выполнять арифметические операции. В то время основными инструментами инженера для вычислений были: логарифмическая линейка и (в лучшем случае) механический арифмометр.

Особенно остро проблема автоматизации вычислений проявилась в военном деле. Началась 2-я мировая война. Чрезвычайно актуальной задачей стало повышение точности стрельбы по кораблям и воздушным целям. Данное обстоятельство послужило толчком к

концентрации усилий ученых по созданию электронных вычислительных машин. Именно в это время математик Н. Винер разработал известные уравнения Винера-Хопфа, которые предназначались для прогнозирования движения воздушных целей.

Существенные успехи в автоматизации вычислений были достигнуты создателями прибора для управления артиллерийским зенитным огнем (ПУАЗО). Первые [приборы для управления артиллерийским огнём](#) были разработаны для дальнобойной [морской артиллерии](#) в конце XIX века. С появлением авиации те же принципы были применены для управления зенитным огнём по высоколетящим целям. Приборы такого типа были разработаны в конце [1930-х](#) годов компаниями [Vickers-Armstrongs](#) (Великобритания), [Sperry](#) (США), [Siemens](#) (Германия) и другими.

В 1942 году профессор электротехнической школы Мура Пенсильванского университета Джон Маучли представил проект (меморандум) «Использование быстродействующих электронных устройств для вычислений», который положил начало созданию первой электронной вычислительной машины ENIAC (Electronic Numerical Integrator and Computer). Около года проект пролежал без движения, пока им не заинтересовалась Баллистическая исследовательская лаборатория армии США, В 1943 году под руководством Маучли и Эккерта были начаты работы по созданию ENIAC. Демонстрация работы машины состоялась 15 февраля 1946 года.

В СССР компьютеры появились позже. К ним можно отнести МЭСМ (Малая Электронная Счетная Машина) разработанную в институте электротехники АН УССР под руководством С.А. Лебедева в 1950 г. К первому поколению относятся и такие машины как БЭСМ, Урал, М-2, Стрела.

Важным этапом развития прикладной математики в СССР явилась организация Института прикладной математики (ИПМ РАН). Организатор института М. В. Келдыш оказал большое влияние на научный стиль института и характер решаемых задач. М. В. Келдыш, президент Академии наук и активный участник космической и ядерной программ, вовлек свой институт в работы по важнейшим практическим проблемам, в которых научная новизна сочеталась с необходимостью сложных расчётов. Так как подобные задачи часто оказывались на стыке научных дисциплин, коллектив института включал математиков, физиков, механиков, а также специалистов по вычислительной технике [5].

Выводы

Таким образом, прикладная математика – наука особого рода, стоящая на грани между точными, гуманитарными и опытными науками, смело применяющая приёмы, выработанные в каждой из этих групп наук, если они оказываются эффективными. Актуальным направлением развития прикладной математики является разработка методов решения тех задач, которые в настоящее время не поддаются решению.

Литература

- 1 Полякова, Т. С. История математики. Европа XVII-начало XVIII вв.. Краткий очерк: учебное пособие [Электронный ресурс] / Т. С. Полякова. – Электрон. текстовые данные. – Ростов-на-Дону : Южный федеральный университет, 2015. – 126 с. / Интернет-ресурс. – Режим доступа: <http://www.iprbookshop.ru/68564.html>.
2. Стройк, Д.Я. Краткий очерк истории математики / Д.Я. Стройк. – Москва: Наука. 1984. – 284 с.
3. Крылов, А.Н. Лекции о приближенных вычислениях: учеб. пособие / А.Н Крылов. – Москва : Гос. изд-во технико-теоретической литературы, 1954. – 401 с.
4. Крылов А.Н. Прикладная математика и техника: доклад на чрезвычайной сессии АП СССР 21 июня 1931 г. [Электронный ресурс] / Интернет-ресурс. – Режим доступа: http://vivovoco.astronet.ru/VV/PAPERS/BIO/KRYLOV/KRYLOV_05.HTM3.
5. Дубнищева, Т. Я. Концепции современного естествознания : учеб. пособие / Т. Я.

Дубнищева. – 10-е изд., стер. – Москва : Академия, 2009. –325 с.

УДК 622.14

ПРИМЕНЕНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ДЛЯ НАХОЖДЕНИЯ МАКСИМАЛЬНОГО КОМПОНЕНТА СОСТАВА ПОРОД ПО ПЛАСТАМ

Прокопенко Е.В., Исаков А.Д

Донецкий национальный технический университет, г. Донецк
кафедра прикладной математики

доцент кафедры прикладной математики, магистрант кафедры информационных технологий
ГОУ ВПО «ДонАУиГС»

E-mail: prokopenko1515@rambler.ru

Аннотация:

Прокопенко Е.В., Исаков А.Д. Применение информационных технологий для нахождения максимального компонента состава пород по пластам. В статье предлагается концепция по применению линейного программирования для создания хронолитологической модели породного отвала, которая позволит выявить интервалы локализации вредных химических компонент на отвале, и дать рекомендации по дальнейшему использованию этих компонент.

Annotation:

Prokopenko E.V., Isakov A.D. The use of information technology to find the maximum component of the composition of rocks by layers The article proposes a concept for the application of linear programming to create a chronolithological model of a waste dump, which will allow identifying the localization intervals of harmful chemical components in the dump, and give recommendations on the further use of these components.

Общая постановка проблемы

В настоящее время внимание промышленности, как в дальнем, так и в ближнем зарубежье вновь обращается к породным отвалам угольных шахт, т.к. отвальные породы могут представлять интерес в качестве источника вторичного минерального сырья. Широкий спектр различных химических соединений и элементов, включающих германий и редкоземельные, а так же глинозём для производства бокситов, железная руда и непосредственно уголь при высокой доступности, низкой цене и практически неограниченных объёмах исходного сырья - породы, делает перспективным указанное техническое направление, тем более что на территории Восточного Донбасса насчитывается более 450 отвалов угольной промышленности (шахт и обогатительных фабрик), а общий объем складированных в отвалах шахт и обогатительных фабрик пород (углеотходов) оценивается в 273 млн. м³.

В отвалах угольных шахт много запасов некоторых металлов, соизмеримых по объёму с природными месторождениями полезных ископаемых, получение которых для Украины будет экономически выгодным, тем более что в настоящее время многие из минеральных ресурсов уже исчерпаны, что является одной из глобальных проблем [1].

Технология формирования породного отвала заключается в насыпке породы на отвал отдельными ярусами. Так как отдельный ярус породного отвала насыпается определенное количество лет, то можно составить карту насыпки отвала за определенный период времени.

Так как каждый пласт имеет свое геологическое строение, то можно составить прогноз тех химических реакций, которые произойдут при соприкосновении тех или иных элементов, содержащихся в различных пластах, то есть заранее выявить неблагоприятные зоны на отдельном ярусе и в целом на отвале [2].

Используя выше сказанное, можно сформулировать задачу о нахождении максимального компонента среди состава пород, входящих в пласт, а также сформировать вектор, позволяющий оценить объем выдачи на отвал породы, содержащий максимальный компонент по пластам.

С точки зрения линейного программирования, данная задача формулируется следующим образом.

Существует функция

$$f(v_1, v_2, \dots, v_m), \quad (1)$$

где

$$\begin{aligned} V_1 &= (x_1^{(1)}, x_2^{(1)}, \dots, x_j^{(1)}, \dots, x_k^{(1)}) \\ V_2 &= (x_1^{(2)}, x_2^{(2)}, \dots, x_j^{(2)}, \dots, x_k^{(2)}) \\ &\dots \\ V_i &= (x_1^{(i)}, x_2^{(i)}, \dots, x_j^{(i)}, \dots, x_k^{(i)}) \\ &\dots \\ V_m &= (x_1^{(m)}, x_2^{(m)}, \dots, x_j^{(m)}, \dots, x_k^{(m)}) \end{aligned} \quad (2)$$

при ограничениях

$$\begin{aligned} g_1(x_1^{(1)}, x_1^{(2)}, \dots, x_1^{(i)}, \dots, x_1^{(m)}) &\leq b_1 \\ g_2(x_2^{(1)}, x_2^{(2)}, \dots, x_2^{(i)}, \dots, x_2^{(m)}) &\leq b_2 \\ &\dots \\ g_j(x_j^{(1)}, x_j^{(2)}, \dots, x_j^{(i)}, \dots, x_j^{(m)}) &\leq b_j \\ &\dots \\ g_k(x_k^{(1)}, x_k^{(2)}, \dots, x_k^{(i)}, \dots, x_k^{(m)}) &\leq b_k \end{aligned} \quad (3)$$

Где $f(v_1, v_2, \dots, v_m)$ - целевая функция или критерий эффективности. В данном случае, вектор f характеризует значение максимального объема по пластам в зависимости от максимального компонента в составе пород пласта, $X = (x_1, x_2, \dots, x_j, \dots, x_k)$ - варьируемые параметры, которые обозначают соответствующую компоненту в составе пород пласта, $g_1(x_1^{(1)}, x_1^{(2)}, \dots, x_1^{(i)}, \dots, x_1^{(m)}) \leq b_1 \dots g_k(x_k^{(1)}, x_k^{(2)}, \dots, x_k^{(i)}, \dots, x_k^{(m)}) \leq b_k$ - функции, которые задают ограничения на компоненты, входящие в состав пород пласта.

Таким образом, используя данные выше приведенных функций, можно составить матрицу размерностью $[1..k, 1..m]$, где k -количество компонент, m -количество пластов. Данная матрица будет иметь вид, представленный ниже.

$$\begin{array}{cccccc}
 V_1 & V_2 & \dots & V_i & \dots & V_m \\
 x_1^{(1)} & x_1^{(2)} & \dots & x_1^{(i)} & \dots & x_1^{(m)} \\
 x_2^{(1)} & x_2^{(2)} & \dots & x_2^{(i)} & \dots & x_2^{(m)} \\
 \dots & \dots & \dots & \dots & \dots & \dots \\
 x_j^{(1)} & x_j^{(2)} & \dots & x_j^{(i)} & \dots & x_j^{(m)} \\
 \dots & \dots & \dots & \dots & \dots & \dots \\
 x_k^{(1)} & x_k^{(2)} & \dots & x_k^{(i)} & \dots & x_k^{(m)}
 \end{array} \quad (4)$$

Для нахождения вектора $V = (V^{(1)}, V^{(2)}, \dots, V^{(m)})$, характеризующего максимальное значение объемов по пластам в зависимости от химического компонента, необходимо вычислить параметры, входящие в вектор, по формуле

$$V^{(i)} = \sum_{i=1}^m x_i^{(i)} \cdot V_i \quad (5)$$

Используя найденные объемы, можно делать вывод о составе компонент, входящих в пласт, то есть, какие компоненты больше содержатся в объеме породы, отсыпанной на отвал, в зависимости от графика проведения подготовительных выработок.

В результате применения данного алгоритма [3] было рассчитано максимальное значение химического компонента среди работающих пластов шахты «Щегловская – Глубокая» шахтоуправления Донбасс. На рисунке 1 приведена таблица с характеристикой состава компонент и значением объемов пород по пластам.

	m3	l1	к8	V	максимальное значение
	82368,75	11804,8	8125,6		
SiO2	40,12	46,14	37,5	4154018	46,14
Fe2O3	32,55	24,33	28,54	3498,742	32,55
CaO	4,65	1,47	4,7	321,2606	4,7
MgO	1,13	0,93	1,25	12,4966	1,25
SO3	3,62	0,77	4,32	10,2067	4,32
Al2O3	14,1	21,22	17,86	144,5366	21,22
TiO2	0,75	0,6	0,47	31,7012	0,75
P2O5	0,15	0,32	0,25	0,422	0,32
K2O	1,28	2,18	1,4	1,2396	2,18
Na2O	1,62	1,67	1,8	8,2342	1,8
MnO	0,06	0	0	0,0972	0,06

Рис.1- Нахождение максимального компонента по пластам

На рисунках 2 показаны диаграммы, характеризующие долю каждого компонента в общем объеме отсыпанной породы на отвал, в зависимости от наименования пласта.

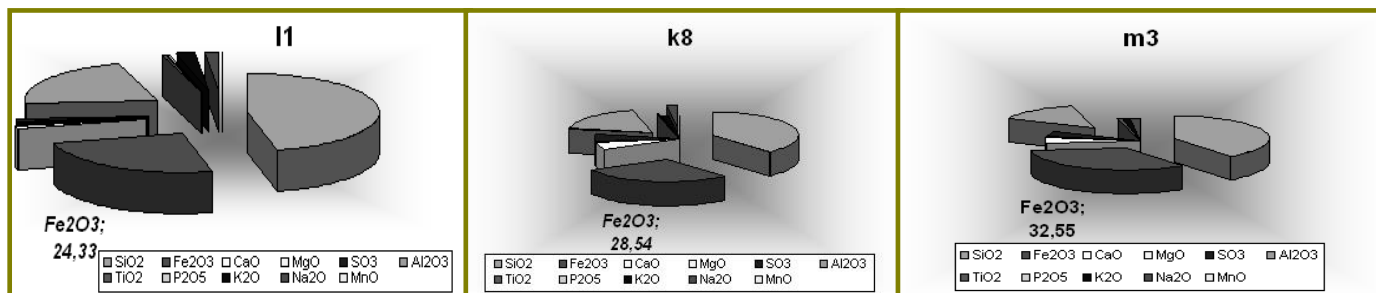


Рис.2- Состав компонент, входящих в отвал по пластам

На основании приведенных диаграмм была замечена закономерность, что все пласты содержат железо (Fe_2O_3), которое необходимо для строительства и для тяжелой промышленности, но доля данного компонента в пластах имеет разное значение. Используя пакет Surfer для построения пространственной модели, на рис.3 построена карта породного отвала в периоде за два года с нанесением мест локализации данного компонента.

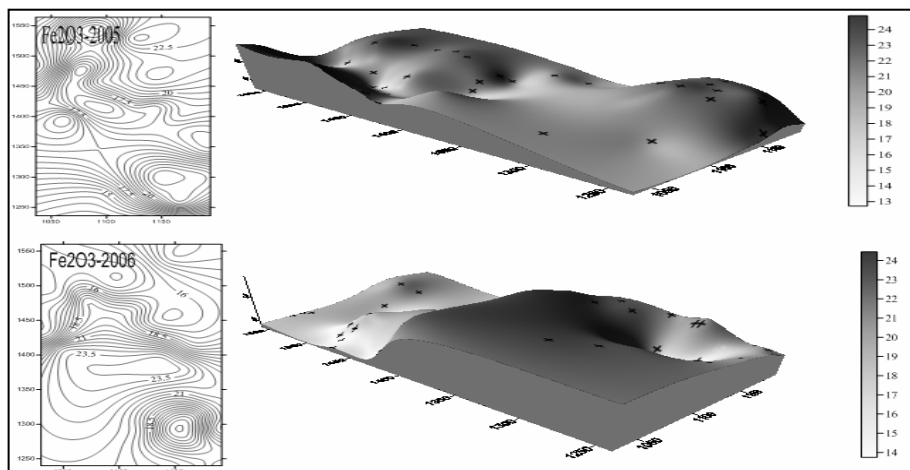


Рис.3- Построение поверхности отвала для компонента Fe_2O_3

Используя аналитическую платформу Deductor, которая применима для решения задач распознавания и обработки данных, таких как парциальная обработка данных (подготовка к анализу) прогнозирование, поиск закономерностей и пр. Платформа применима в задачах, где требуется консолидация и отображение данных различными способами, построение моделей и последующее применение полученных моделей к новым данным можно провести кластеризацию данных компонент на основе нейронных сетей.

Результатом данной кластеризации является карта с нанесением раскраски по каждому компоненту и расчетом статистики по каждому кластеру.

Полученная объективная информация может быть использована для выработки дальнейших исследований и рекомендаций по насыпке породы на отвал. Современные системы управления требуют выработки обоснованных управленческих решений.

Аналитические платформы позволяют принимать обоснованные и эффективные решения, которые выводят возможности анализа данных на совершенно новый уровень, позволяющий каждому сотруднику получать релевантную информацию, необходимую для развития бизнеса.

Аналитические инструменты позволяют пользователю для достижения лучших бизнес-результатов использовать функции создания отчетов, анализа, моделирования,

планирования и совместной работы для реализации инициатив по бизнес-аналитике, управлению производительностью и анализу

На рис.4 показан вид экрана работы системы Deductor для данной задачи.

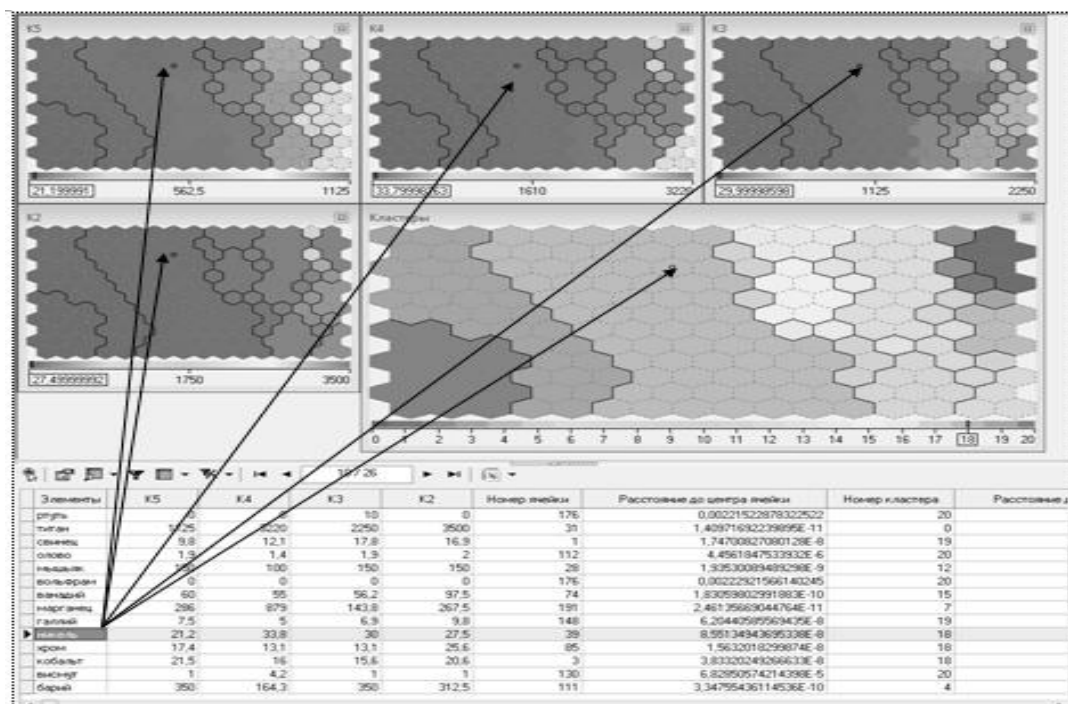


Рис.4.-Фрагмент листа платформы Deductor с кластеризацией вредных элементов по пластам
Выводы. Таким образом, зная, что происходит с насыпкой породы в пределах одного пласта, и, зная какие работают пласты, то ли одновременно, то ли каждый по отдельности, можно составить хронолитологическую модель насыпки породы, в результате которой могут быть выявлены места локализации конкретных химических компонент. По данной модели можно составить рекомендации контроля над данными компонентами, а также дать рекомендации по дальнейшему использованию этих веществ. [4,5]

Литература

1. Панов Б.С., Проскурня Ю.А. Новые виды минерального сырья Донбасса// Матеріали науково-практичної конференції//Донбасс-2020: наука і техніка-виробництво Донецьк: ДонНТУ.-2002, с.74-77.
2. Прокопенко Е.В. Разработка динамической модели породных отвалов угольных шахт/ С.В.Борщевский, Е.В.Прокопенко// Наукові праці УкрНДМІ НАН України. Випуск 6 / під заг. Ред. А.В. Анциферова. - Донецьк, УкрНДМІ НАН України, 2010. – С. 14-19.
3. Прокопенко Е.В. //Борщевский С.В., Прокопенко Е.В.// Разработка алгоритма для нахождения максимального компонента в составе пород на отвале. Материалы Международной научно-практической конференции молодых ученых и студентов ТулГУ, Тула, 2011.- 117с
4. Вахмянин И.С. Разработка модели алгоритма управления информационными потоками в ситуационных центрах органов государственной власти / Вахмянин И.С., Ильин Н.И., Новикова Е.В.—Бизнес-Информатика №1(15)- 2011.—С. 3—9.
5. Прокопенко Е.В. Построение пространственной модели отвала с использованием программного пакета Surfer /С.В.Борщевский, Е.В.Прокопенко// Збірник наукових праць НГУ. — Національний гірничий університет, 2010. —№34, т.1.— С. 82—87.

УДК 519.6 (076)

ТЕОРИЯ АЛГОРИТМОВ КАК ОДИН ИЗ НЕОБХОДИМЫХ ИНСТРУМЕНТОВ ПРОГРАММИСТА

Канатуш С.В.¹, Ефименко К.Н.²

Донецкий национальный технический университет
студент гр. ИС-17а¹, кафедра прикладной математики²

E-mail: kanatush.lorraine01@gmail.com

Аннотация:

Канатуш С.В., Ефименко К.Н. Теория алгоритмов как один из необходимых инструментов программиста. Рассмотрено значение теории алгоритмов как базовой части информатики и программирования, необходимых при обучении специалистов в области IT-технологий. Показаны примеры рационального использования алгоритмов.

Annotation:

Kanatush S.V., Efimenko K.N. Theory of algorithms as one of the necessary programmer's tools. The value of the theory of algorithms as the basic part of computer science and programming, necessary for training specialists in the field of IT-technologies, is considered. Showing examples of rational use of algorithms.

Введение

После поступления в образовательное учреждение высшего профессионального образования, будущим программистам предстоит изучить множество дисциплин, которые, по мнению некоторых студентов, не являются обязательными для полноценного формирования грамотного специалиста в области современных IT-технологий. Такими «необязательными» дисциплинами многие считают, например, численные методы, схемотехнику, структуры данных, теорию алгоритмов. Споры не утихают даже над последней из перечисленных дисциплин, хотя понятие «алгоритм» имеет, если не первостепенное, то ключевое значение в бурно развивающейся отрасли.

Таким образом, мы подходим к вопросу: необходимы ли действительно специалистам в области информационных технологий знание этих дисциплин и теории алгоритмов в частности? По мнению авторов, образование должно охватывать как можно большую аудиторию и давать максимально возможный набор навыков. Программист в первую очередь должен уметь анализировать и решать проблемы – это основной навык, которым должны овладеть выпускники факультета компьютерных наук.

Целью данной работы является обоснование необходимости изучения теории алгоритмов для становления грамотного и конкурентноспособного IT-специалиста.

Теория алгоритмов – основа информатики и программирования

Теория алгоритмов – наука, находящаяся на стыке математики и информатики, это часть математической логики, изучающая общие свойства и закономерности алгоритмов, а так же разнообразные формальные модели и их представление. Теория алгоритмов предназначена для исследования проблемы разрешимости и обоснования математики. Все вопросы, касающиеся теории алгоритмов, можно условно разбить на четыре категории [1].

1. Вопросы, относящиеся к теории вычислимости (какие функции являются вычислимыми, какие нет).

2. Вопросы, относящиеся к теории сложности, т. е. вопросы, которые позволяют выяснить, за какое время можно вычислить ту или иную функцию, а также вопросы поиска

оптимального решения.

3. Вопросы, относящиеся к программированию, т. е. реализации алгоритма.

4. Вопросы, относящиеся к методам вычислений, т. е. вопросы о нахождении приближенных решений уравнений, неравенств и их систем (численные методы).

Многие студенты желают изучать только современные инструменты разработки программного обеспечения, но изучать теорию – это наиболее оптимально с точки зрения получения качественного образования. Так как приобретенные знания и навыки можно применить в любой области, а библиотеки или фреймворки быстро устаревают и, следовательно, их изучение не представляет такой ценности. Кроме того, имея хорошую базу знаний, изучить подобные инструменты не составит большого труда.

Также стоит отметить, что рациональное использование библиотек требует их глубокого понимания так как, использование функций из библиотек может давать побочные эффекты или непредсказуемые результаты, которые вы не сможете оценить без понимания сути алгоритмов. Не говоря уже о том, что различным инструментам и библиотекам часто необходимо указывать какие алгоритмы, структуры данных или технологии будут использоваться. Без элементарных знаний придется либо идти читать спецификацию, либо выбирать наугад.

Довольно часто для решения конкретных задач недостаточно средств, предлагаемых библиотеками и фреймворками, поэтому возникает необходимость в написании собственных функций и модулей, что требует знаний о структурах данных и об алгоритмизации.

Следует отметить, что при изучении программирования, одному из авторов на лабораторных работах приходилось реализовывать некоторые алгоритмы и структуры, которые впоследствии он использовал в других проектах в качестве подключаемых модулей, что является очень удобным, учитывая, что автор абсолютно точно знал их поведение и способ реализации.

На самом деле знания по теории алгоритмов применяются всеми программистами каждый день, просто мы привыкли к этим вещам настолько, что даже не задумываемся над этим. Если вы спросите программиста, задумывался ли он о происхождении таких типов алгоритмов как следование, ветвление, цикл и функция, то ответ будет отрицательным. Таким образом, какую бы задачу вы не решали – будь то сайт с выборкой информации из базы данных, или скрипт, работающий на стороне сервера, вы будете использовать структуры, которые первоначально были описаны не языком программирования, а теорией. Следовательно, нельзя недооценивать роль теории алгоритмов при разработке языков программирования такими, какими они являются на сегодняшний день.

Нельзя не отметить, что теория алгоритмов содержит раздел, который позволяет проводить оценку сложности алгоритмов. При этом сложность алгоритмов обычно оценивают по времени его выполнения или по объему используемой памяти. Знание этого раздела позволяет программисту грамотно определять и распределять необходимые вычислительные ресурсы при работе программного обеспечения.

Например, самый простой алгоритм сортировки пузырьком имеет два цикла по n итераций, соответственно имеет сложность n^2 . Но эта оценка является приближенной и оценивает самый неоптимизированный алгоритм. Проверим скорость работы алгоритма сортировки пузырьком с оптимизацией и без. Проверка будет осуществляться на языке программирования C++ с массивом на десять тысяч элементов.

Следующий алгоритм выполняется за 1.355 секунды:

```
for (int i = 0; i < 10000; i++)  
{  
    for (int j = 0; j < 10000-1; j++)  
    {  
        if (mas[j] > mas[j + 1])
```

```

        {
            swap(mas[j], mas[j + 1]);
        }
    }
}

```

А после оптимизации алгоритм выполняется за 1.131 секунды:

```

int i, j;
bool flag;
do
{
    flag = false;
    for (j = 0; j < 10000-1; j++)
    {
        if (mas[j] > mas[j + 1])
        {
            swap(mas[j], mas[j + 1]);
            flag = true;
        }
    }
} while (flag);

```

Таким образом, выигрыш в скорости составил более 16.5%.

Значит перед программистом во время написания программы стоит две задачи: написать не только работоспособный код, но и сделать его оптимальным, т.е. написать его так, чтобы для выполнения программы требовалось как можно меньше вычислительных мощностей компьютера. Некоторые считают современные компьютеры достаточно мощными, что позволяет опустить вторую задачу, однако это далеко не так. Для анализа трудоемкости алгоритма приведена таблица, показывающая время выполнения программы, реализующей алгоритм определённой сложности в зависимости от размера входных данных при скорости 106 операций в секунду (табл.1).

Таблица 1 – Зависимость времени выполнения программы от сложности алгоритма и размера входных данных

Сложность алгоритма	Размер исходных данных					
	10	20	30	40	50	60
n	0,000001 с.	0,000002 с.	0,000003 с.	0,000004 с.	0,000005 с.	0,000005 с.
n ²	0,0001 с.	0,00004 с.	0,0009 с.	0,0016 с.	0,0025 с.	0,0036 с.
n ³	0,0001 с.	0,008 с.	0,027 с.	0,064 с.	0,125 с.	0,216 с.
n ⁵	0,1 с.	3,2 с.	24,3 сек.	1,7 мин.	5,2 мин.	13 мин.
2 ⁿ	0,0001 с.	1 с.	17,9 мин.	12,7 дней	35,7 век.	366 век.
3 ⁿ	0,059 с.	58 мин.	6,5 лет	3855 веков	2·10 ⁸ век.	1,3·10 ¹¹ в.

Из таблицы видно, что время выполнения программы зависит от сложности выполнения алгоритма и размера обрабатываемых данных и может значительно отличаться. Таким образом, программист должен быть «жадным» к вычислительным мощностям и

стараться их минимизировать [2].

Нельзя не отметить, что появление точного понятия алгоритма позволило сформулировать алгоритмически не разрешимые проблемы, то есть задачи, для решения которых невозможно построить алгоритм. Задача называется алгоритмически неразрешимой, если не существует машины Тьюринга, рекурсивной функции или нормального алгоритма Маркова, которые ее решают. Например, неразрешимой оказалась проблема распознавания эквивалентности алгоритмов: нельзя построить алгоритм, который по любым двум алгоритмам или программам, выяснял бы, вычисляют они одну и ту же функцию или нет.

Знание основных неразрешимостей теории алгоритмов необходимо для IT-специалиста. Оно предостережет его от увлечения алгоритмизацией точно так, как знание основных законов физики предостерегает от попыток создания вечного двигателя [3].

Также стоит сказать, что даже на сегодняшний день на собеседованиях на должности разработчиков интервьюеры многих IT-компаний просят показать знания, казалось бы, простейших алгоритмов: бинарный поиск, шейкерная сортировка и сортировка вставками, но это имеет смысл, ведь если человек не может реализовать стандартный алгоритм, то как он будет решать нетривиальные задачи? При этом опытные программисты говорят, что специалист должен уметь выводить алгоритмы, а не знать их. Ровно, как и математик должен уметь выводить доказательства.

Выводы

Можно ли программисту обойтись без знаний теории алгоритмов и непосредственно алгоритмов? К сожалению – можно, и таких «программистов» очень много. Только назвать их программистами довольно-таки сложно. Исходя из статистики, такие специалисты не задерживаются в своих местах работы больше года. Как правило, они обладают знанием небольшого набора задач, которые они могут решать, и если сделать шаг в сторону, то человек теряется и ему нужно обучать себя новым навыкам [4].

Попросив совета у более опытного коллеги о том, как улучшить профессиональные навыки, велик шанс того, что его мнение будет звучать следующим образом: одним из главных факторов роста IT-специалиста является совершенствование знаний структур данных и алгоритмов, а также, что самое главное, совершенствование навыка их использования на практике. И, что очевидно не сразу, но это один из немногих верных ответов на данный вопрос. Алгоритмов, равно как и структур данных, существует большое множество и знания основных из них является обязательным, если вы действительно хотите стать профессионалом в сфере IT-технологий.

Таким образом, освоение теории алгоритмов необходимо для эффективного использования возможностей современной компьютерной техники, изучения программирования и информатики. Знание основ теории алгоритмов необходимо практически в любой современной научно-исследовательской работе и проектировании.

Литература

1. Алферова, З.В. Теория алгоритмов / З.В. Алферова. – Москва : Статистика, 1973. – 164 с.
2. Черняк, А. Методы оптимизации: теория и алгоритмы / А. Черняк. – Санкт-Петербург : Юрайт, 2016. – 357 с.
3. Стивенс, Р. Алгоритмы. Теория и практическое применение / Р. Стивенс. – Москва : Эскмо, 2016. – 544 с.
4. Луридаc, П. Алгоритмы для начинающих. Теория и практика для разработчика / П. Луридаc. – Москва : Эскмо, 2017. – 608 с.

УДК 621.941

ЗАДАЧИ ОПТИМИЗАЦИИ РАСПИСАНИЯ ПРОГРАММ ОБРАБОТКИ ДЕТАЛЕЙ ДЛЯ СТАНКОВ С ЧПУ

Кононенко Д. Н., Лазебная Л.А., Бельков Д.В.
Донецкий национальный технический университет
кафедра прикладной математики

E-mail: konon966@gmail.com, l_lazebnay@mail.ru, belkovdv@list.ru

Аннотация:

Кононенко Д.Н., Лазебная Л.А., Бельков Д.В. Задачи оптимизации расписания программ обработки деталей для станков ЧПУ. В статье рассматривается практическая задача, возникающая в САПР для токарной обработки деталей, которая заключается в оптимизации программного графика. Предложены два приближенных алгоритма решения задачи. Целью первого является - минимизация переналадок. Целью второго – минимизация вероятности сбоя оборудования при обработке деталей. Алгоритмы могут практически применяться для составления расписания программ обработки деталей для станков с ЧПУ.

Annotation:

Kononenko D.N., Lazebnaya L.A., Belkov D.V. The task of optimizing the schedule of parts processing programs for CNC machines. The article discusses the practical problem arising in CAD for turning parts, which is to optimize the software schedule. Two approximate algorithms for solving the problem are proposed. The purpose of the first is to minimize readjustments. The purpose of the second is to minimize the probability of equipment failure when processing parts. Algorithms can be practically applied for scheduling part machining programs for CNC machines.

Введение

Использование на предприятиях станков с числовым программным управлением (ЧПУ) намного ускоряет производство и снижает себестоимость продукции. Станок вместе с сопровождающим роботом автоматически выполняет обработку серии деталей подобного вида. Роль человека сводится только к созданию программ и переналадкам станка при переключениях от одной программы к другой.

Для повышения эффективности программирования станка в работе [1,2] предложена система автоматизированного проектирования (САПР) технологических процессов токарной обработки деталей. Ее применение позволяет:

- проследить ход обработки детали;
- уменьшить количество рутинной работы, связанной с повторяющимися действиями;
- уменьшить количество ошибок в программе для ЧПУ;
- автоматизировать создание мнемкокода программы для ЧПУ.

При составлении программы разработчик может следить за обработкой детали, используя графическое изображение траектории инструмента. Просмотр опорных точек детали позволяет разработчику точно задавать параметры технологических операций.

Предложенная в работе [3] САПР внедрена на Орловском заводе УВМ и на Северодонецком НПО “Импульс”. Ее использование при программировании станков приводит к значительному повышению производительности труда за счет ускорения процесса создания программ.

Однако данная система не дает возможности увеличить скорость обработки партии деталей за счет выполнения программ в оптимальной последовательности. Для достижения этой цели необходимо решить задачу составления рационального расписания выполнения программ. Она возникает на этапе эксплуатации станков с ЧПУ при механической обработке деталей. В данной работе предлагаются две постановки задачи составления рационального расписания выполнения программ и методы их решения.

В первой постановке задача состоит в минимизации количества переналадок станков с целью ускорения обработки партии деталей. Исходными данными задачи являются сроки выполнения программ и директивное время обработки деталей. Функциональное назначение программ предполагается заданным в соответствии с технологией обработки деталей.

Во второй постановке задача состоит в максимизации вероятности обработки партии деталей без сбоев оборудования, которые задерживают выполнение программ обработки деталей. Ускорение выполнения программ может быть достигнуто за счет составления расписания программ с максимизацией вероятности их выполнения без сбоев оборудования.

Задача минимизации количества переналадок станков

Обозначим: m - число деталей; n - число программ, совпадающее с максимальным количеством переналадок станка; T_i - директивное время обработки детали i ; $x_j = 1$, если программа j используется при обработке деталей, иначе $x_j = 0$; B_j - время выполнения программы j ; U_j - остаточное время выполнения программы j : $U_j = B_j - \sum_{i=1}^m V_i x_j$; $a_{ij} = 1$, если деталь i может обрабатываться программой j : $T_i \geq U_j$, иначе $a_{ij} = 0$.

В задаче составления расписания необходимо минимизировать количество используемых программ, что приводит к минимизации числа переналадок. Задача изоморфна известной [4] задаче о покрытии.

Целевая функция:

$$\sum_{j=1}^n x_j \rightarrow \min \quad (1)$$

Ограничения:

$$\sum_{j=1}^n a_{ij} x_j \geq 1, \quad i = 1, 2, \dots, m \quad (2)$$
$$X_{ij} \in \{0,1\}$$

В задаче необходимо найти матрицу A и вектор X , которые обеспечивают минимум целевой функции (1) при ограничениях (2).

Метод решения задачи минимизации количества переналадок станков

Для решения задачи (1)-(2) можно использовать “жадный” (greedy) метод с временной сложностью $O(mn)$:

Шаг 1. Переменной i присвоить значение единица, элементам вектора U присвоить значения элементов вектора B , элементам матрицы A и вектора X присвоить значение ноль.

Шаг 2. Переменным a_{ij} и x_j присвоить значение единица, если $V_i \geq U_j$.

Шаг 3. Переменной i присвоить значение $(i + 1)$.

Шаг 4. Если $i \leq m$ то перейти к шагу 2, иначе перейти к шагу 5.

Шаг 5. Вычислить значение целевой функции и завершить метод.

Для жадных методов в задаче о покрытии известна верхняя оценка максимальной относительной погрешности: $A/P \leq 1 + \ln(m)$, где A - решение, полученное “жадным” методом, P - оптимальное решение задачи. Жадный метод позволяет получить оценку значения целевой функции через параметры матрицы A : если $\varepsilon(A)$ - глубина матрицы A , и каждый столбец матрицы A содержит не менее k единиц, то справедливо неравенство: $\varepsilon(A) \leq n/k(1 + \ln(mk/n)) + 1$. Для любого фиксированного числа $\delta > 0$ для большинства матриц выполняется условие: $\varepsilon(A) \geq (1 - \delta)n/k \cdot \ln(mk/n)$.

Таким образом, жадный метод для задачи о покрытии является асимптотически оптимальным для широкого класса матриц A [4].

Задача максимизации вероятности обработки деталей без сбоев оборудования

Обозначим: T - время выполнения программы, P - вероятность того, что сбой оборудования не мешает выполнению программы. Если в процессе выполнения программы возник сбой оборудования, то ее выполнение прекращается. Программа будет выполняться заново после устранения неполадки. Поэтому величина P есть вероятность выполнения программы с первой попытки.

Определим значения P при заданной величине T и условии, что поток сбоев – это поток Эрланга k -го порядка с параметром λ . Пусть, например, программа начинает выполняться сразу после устранения очередного сбоя. В этом случае вероятность P определяется по формуле $P = 1 - F(T)$, где $F(T) = 1 - \sum_{n=0}^{k-1} (\lambda \cdot T)^n \cdot e^{-\lambda T} / n!$ [3]. Если $k=1$, то

$F(T) = 1 - e^{-\lambda T}$ и $P = e^{-\lambda T}$. В этом случае значения P будут тем больше, чем меньше величина T . Если $k=2$, то $F(T) = 1 - \lambda \cdot T \cdot e^{-\lambda T}$, $P = \lambda \cdot T \cdot e^{-\lambda T}$, $dP/dT = \lambda \cdot e^{-\lambda T} - \lambda^2 \cdot T \cdot e^{-\lambda T}$. Функция P имеет максимальное значение, если $T = 1/\lambda$.

Пусть программа начинает выполняться в момент, когда после очередного сбоя оборудования прошло t единиц времени. Это означает, что программа не является первой в списке выполняемых программ. Перед ней были другие программы, на выполнение которых потребовалось t единиц времени. В этом случае значение P определим по формулам [5]:

$$f_r = (\lambda[\lambda(r+t)]^{k-1} \cdot e^{-\lambda(r+t)}) / ((k-1)! \cdot R((k-1), \lambda \cdot t)), \text{ где}$$

$$R((k-1), \lambda \cdot t) = \sum_{n=0}^{k-1} (\lambda \cdot t)^n \cdot e^{-\lambda t} / n!; \quad P = \int_T^{\infty} f_r dr$$

Для быстрого выполнения программ необходимо составить их оптимальное расписание. Критерием оптимальности является максимизация вероятности выполнения программ с первой попытки (без сбоев оборудования).

Обозначим: T_i - время выполнения программы i , P_{ij} - вероятность выполнения программы i с первой попытки, если в расписании оно находится на j -м месте. Пусть на j -м месте может находиться несколько программ, т.е. разные программы могут начинать выполняться одновременно, но на разных станках. В таком случае количество мест в расписании меньше, чем число программ.

Обозначим: m - количество программ, n - количество мест в расписании, $X_{ij} = 1$, если задание i должно быть в расписании на j -м месте, иначе $X_{ij} = 0$. Пусть число программ, которые могут начинать выполняться одновременно ограничено и B_j - допустимое суммарное время выполнения программ, которые находятся в расписании на j -м месте. Задача об оптимальном расписании выполнения программ имеет следующий вид.

Целевая функция:

$$\sum_{i=1}^m \sum_{j=1}^n P_{ij} X_{ij} \rightarrow \max \quad (3)$$

Ограничения:

$$\begin{aligned} \sum_{j=1}^n X_{ij} &= 1, \quad i = 1, 2, \dots, m \\ \sum_{i=1}^m T_i X_{ij} &\leq B_j, \quad j = 1, 2, \dots, n \\ X_{ij} &\in \{0,1\} \end{aligned} \quad (4)$$

В задаче необходимо найти матрицу X расписания программ, обеспечивающую максимум целевой функции (3) при ограничениях (4). В задаче максимизируется вероятность выполнения программ без сбоев оборудования.

Для решения задачи (3)-(4) в данной работе предлагаются два жадных метода, отличающиеся друг от друга по точности и времени решения задачи.

Методы решения задачи максимизации вероятности обработки деталей

Методы решения задачи (3)-(4) состоят из mn этапов, m - количество программ, n - количество мест в расписании. На i -м этапе выполняется процедура назначения i -й программы на одно из мест в расписании.

Процедура состоит из 3 шагов.

На первом шаге метода A1 определяется место в расписании с наибольшим значением P_{ij} , $j = 1, \dots, n$. На втором шаге, если условие $\sum_{i=1}^m T_i X_{ij} \leq B_j$ выполняется, то программа назначается на это место. Иначе назначение программы на найденное место запрещается, и процедура повторяется с первого шага. Временная сложность метода A1 составляет $O(n^2 \cdot m)$.

На первом шаге метода A2 сравниваются величины B_j и величины $\sum_{i=1}^m T_i X_{ij}$, чтобы найти допустимые для программы места в расписании. На втором шаге, среди найденных мест в расписании определяется место с наибольшим значением P_{ij} , $j = 1, \dots, n$. На третьем шаге программа назначается на это место. Исходными данными для процедуры служат значения P_{ij}, T_i, B_j . Временная сложность метода A2 составляет $O(mn)$.

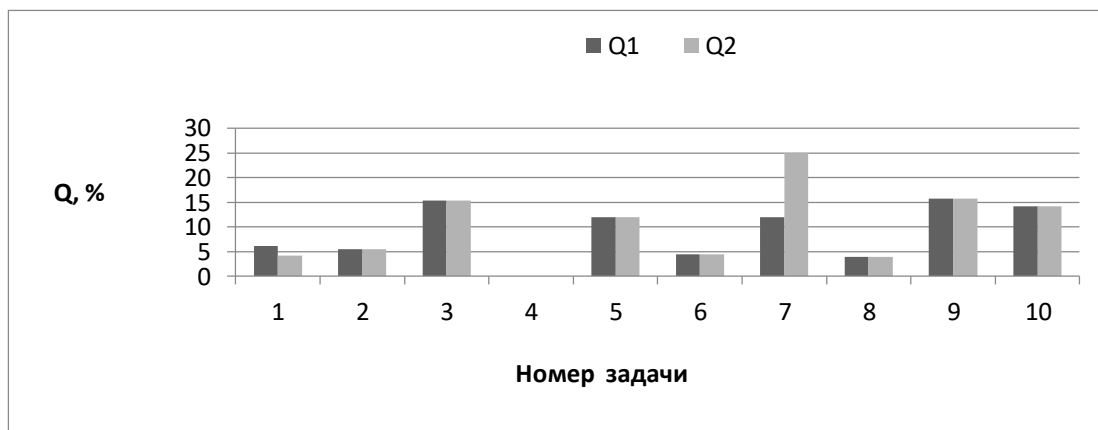


Рис. 1. Относительная погрешность методов A1, A2

Вычислительные эксперименты при исследовании предложенных методов показали, что в большинстве случаев метод А2 превосходит метод А1 по точности. Результаты экспериментов показаны на рисунке 1. Однако в некоторых случаях метод А2 значительно уступает методу А1 по быстродействию. Поэтому среди предложенных методов решения задачи невозможно выбрать лучший по всем параметрам.

При решении конкретных задач составления расписаний можно использовать последовательно оба метода с целью выбора расписания, обеспечивающего максимальное значение целевой функции.

Заключение

Увеличение скорости обработки партии деталей на станке с ЧПУ может быть достигнуто за счет выполнения программ в оптимальной последовательности. Поэтому на этапе эксплуатации станков с ЧПУ при механической обработке деталей возникает задача составления рационального расписания выполнения программ. В данной работе предлагаются две постановки задачи и методы их решения.

В первой постановке задача состоит в минимизации количества переналадок станков с целью ускорения обработки партии деталей. Исходными данными задачи являются сроки выполнения программ и директивное время обработки деталей. Функциональное назначение программ предполагается заданным в соответствии с технологией обработки деталей. Задача минимизации количества переналадок станков сформулирована в виде комбинаторной задачи о покрытии. В статье предложен эффективный жадный метод ее решения. Он позволяет в большинстве случаев получать решения задачи близкие к оптимальным.

Сбои оборудования задерживают выполнение программ механической обработки деталей на ЧПУ. Ускорение выполнения программ может быть достигнуто за счет составления расписания программ с максимизацией вероятности их выполнения без сбоев оборудования. В работе предложены два метода составления такого расписания и определены их трудоемкости.

Предложенные в статье методы позволяют повысить производительность станков с ЧПУ за счет сокращения количества переналадок и повышения вероятности выполнения программ без сбоев оборудования. Эти методы необходимо использовать в системах автоматизированного проектирования технологических процессов механической обработки деталей. Это позволит повысить эффективность процессов обработки деталей на этапе эксплуатации станков.

Литература

1. Основы программирования и наладки станков с ЧПУ: учебное пособие / В.П. Должиков. – Томск: Изд-во Томского политехнического университета, – 2011. – 143 с.
2. Программирование технологических процессов на станках с программным управлением : учеб. пособие / В.С. Мычко. – Минск : Выш.шк., 2010. – 287 с.
3. Фроликов Е.А., Андросова С.И. САПР ТП механической обработки деталей на станках с ЧПУ. //Вісник східноукраїнського національного університету. - 2002. - № 4. - С. 125-129.
4. Новиков Ф.А. Дискретная математика для программистов: Учебник для вузов. – 3 - изд. - СПб.: Питер, 2009. - 384 с.
5. Лифшиц М.А. Случайные процессы от теории к практике: Учебное пособие. - СПб.: Лань, 2016. - 320 с.

УДК 004.04

МОДЕЛЬ БАКА-СНЕППЕНА ДЛЯ ТРАНСПОРТНОГО ЗАТОРА

Пауков Д.В., Зензеров В.И., Бельков Д.В.

Донецкий национальный технический университет
кафедра прикладной математики

E-mail: belkovdv@list.ru, mpvik@3g.ua, denshik591@gmail.ru

Аннотация

Пауков Д.В., Зензеров В.И., Бельков Д.В. Модель Бака-Снеппена для транспортного затора. Анализ теоретических подходов к построению моделей транспортных потоков и проблем, связанных с несоответствием результатов моделирования и результатов эмпирических исследований, позволило выявить недостатки существующих подходов. В статье предлагается новый подход к моделированию возникновения транспортных заторов с использованием алгоритма Бака-Снеппена, который позволит в большей мере добиться соответствующих реальности математических моделей для изучения поведения транспортных потоков. Исследование модели выполнено в среде Scilab.

Ключевые слова: транспортный затор, само организованной критичности, генератор розового шума, алгоритм Бака-Снеппена.

Abstract

Paukov D.V., Zenzerov V.I., Belkov D.V. The Buck-Sneppen model for traffic congestion. An analysis of theoretical approaches to the construction of traffic flow models and problems associated with the discrepancy between the simulation results and the results of empirical research allowed us to identify the shortcomings of the existing approaches. The article proposes a new approach to modeling the occurrence of traffic congestion using the Buck-Sneppen algorithm, which will make it possible to achieve more relevant mathematical models for studying the behavior of traffic flows. The research is executed in the Scilab environment.

Keywords: transport congestion, self-organized criticality, generator of pink noise, the algorithm of the Buck-Sneppen.

Общая постановка проблемы

Транспортный затор – это скопление на дороге транспортных средств, движущихся со средней скоростью, значительно меньшей, чем нормальная скорость для данного участка дороги. При образовании затора значительно снижается пропускная способность участка дороги. Если прибывающий поток транспорта превышает пропускную способность участка дороги, затор растёт лавинообразно. Дорожные заторы распространены по всему миру как результат увеличивающейся автомобилизации и урбанизации, так и увеличивающейся плотности заселения территории. Они уменьшают эффективность дорожно-транспортной инфраструктуры, увеличивая время в пути, расход топлива и уровень загрязнения окружающей среды. В условиях затора резко возрастает вероятность дорожно-транспортного происшествия (ДТП). Ограничение и регулирование интенсивности движения может влиять на количество ДТП [1].

Заторы могут появляться без особой причины. Случайное незначительное снижение скорости одного автомобиля оказывается достаточным, чтобы выстроилась большая пробка. Пробки имеют фрактальный характер: внутри больших пробок есть пробки меньшего размера. В результате возникает движение в режиме «старт-стоп».

Известно, что дорожное движение на скоростной магистрали является примером розового шума. Он возникает из-за масштабной инвариантности лавин в самоорганизованно-критической системе. В случае дорожного движения этот шум служит математическим описанием непредсказуемой езды рывками в дорожных пробках [2].

Одной из простых моделей генерации розового шума является модель Бака-Снеппена. Первоначально она была предложена для моделирования эволюции. Затем данная модель применялась для объяснения многих реальных процессов [3,4].

Целью статьи является повышение эффективности дорожно-транспортной инфраструктуры за счет математического моделирования возникновения транспортного затора на основе модели Бака-Снеппена. В работе решаются задачи построения и исследования этой модели. Приведены результаты вычислительного эксперимента, выполненного в среде Scilab.

Исследования

Модель Бака-Снеппена представляет собой кольцо элементов. В первоначальный момент времени каждому из элементов назначается в качестве параметра случайное число, например, в диапазоне от 0 до 1. Затем определяется элемент с минимальным параметром. Это «слабое звено» системы и его параметр заменяется новым случайным значением. Кроме этого, заменяются новыми случайными значениями параметры двух непосредственных соседей «слабого» элемента. В этом заключается один цикл работы модели. После большого числа итераций система приходит в состояние динамического равновесия (критическое состояние), которое иногда может нарушаться. Такое прерывистое равновесие называется перемежаемостью. Оно заключается во вспышках высокой активности, прерывающих состояние относительного покоя, когда ее уровень низок или равен нулю. Причем в длительности интервалов между вспышками и в их амплитуде не прослеживаются никаких явных правил. Прерывистое равновесие характерно для многих областей, таких как биологическая эволюция, гидродинамика, сейсмология, экономика [5].

Блок-схема предложенного алгоритма реализации модели Бака-Снеппена показана на рис. 1.

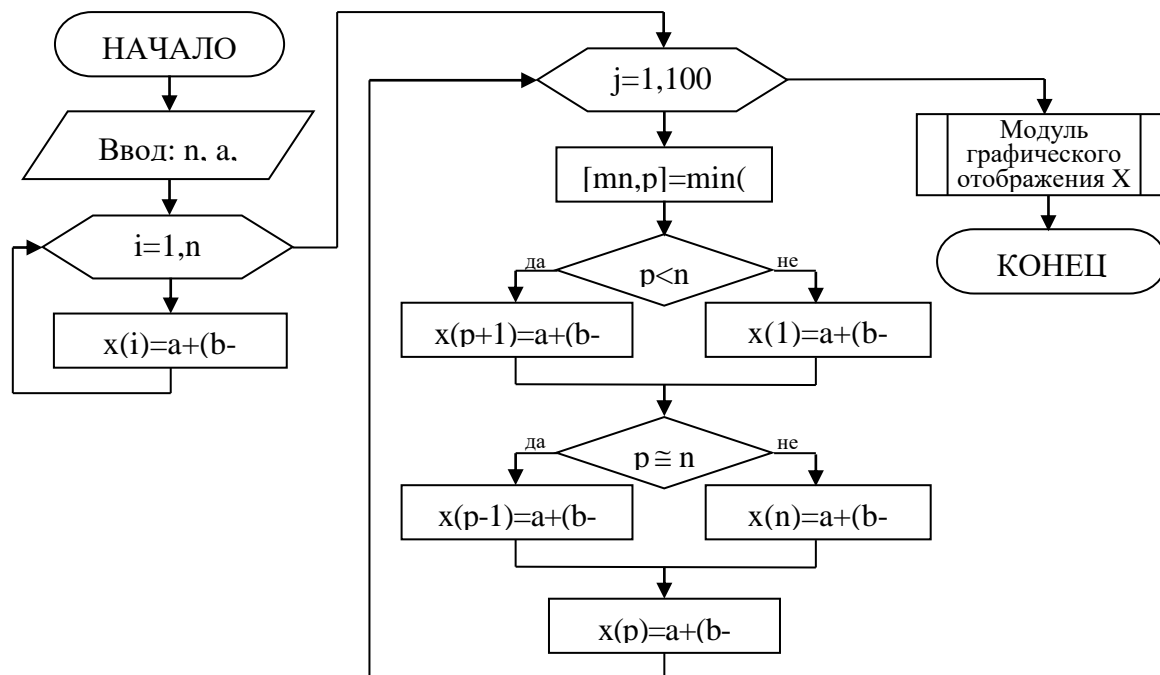


Рисунок 1 – Блок-схема алгоритма Бака-Снеппена
Результаты вычислительного эксперимента показаны на рис. 2–6.

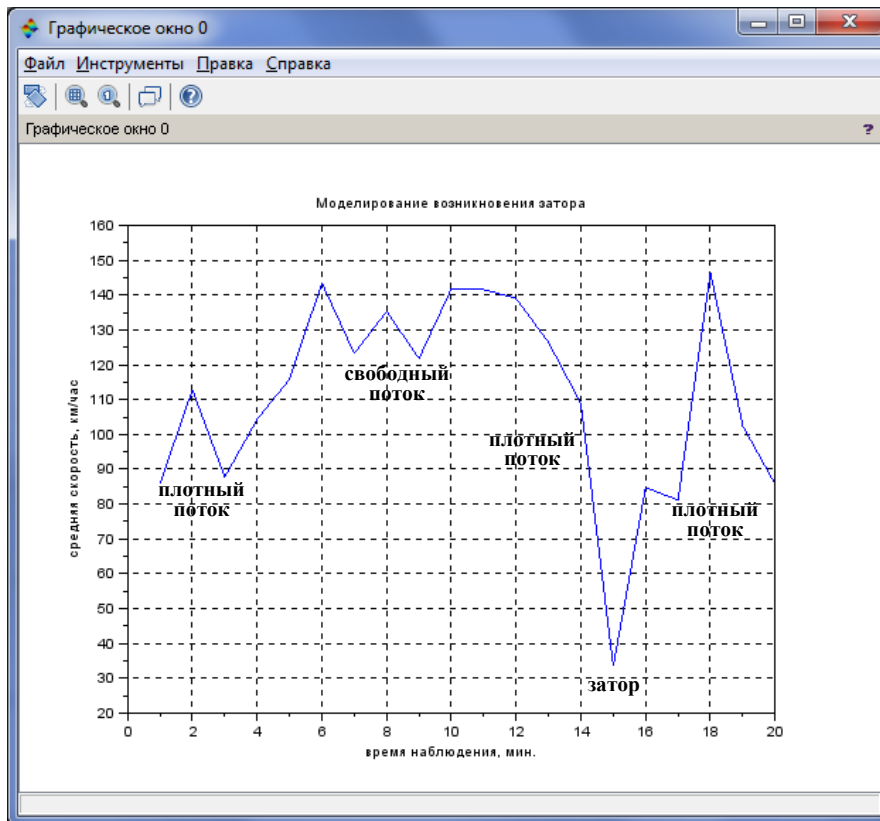


Рисунок 2 – Изучаемый динамический ряд

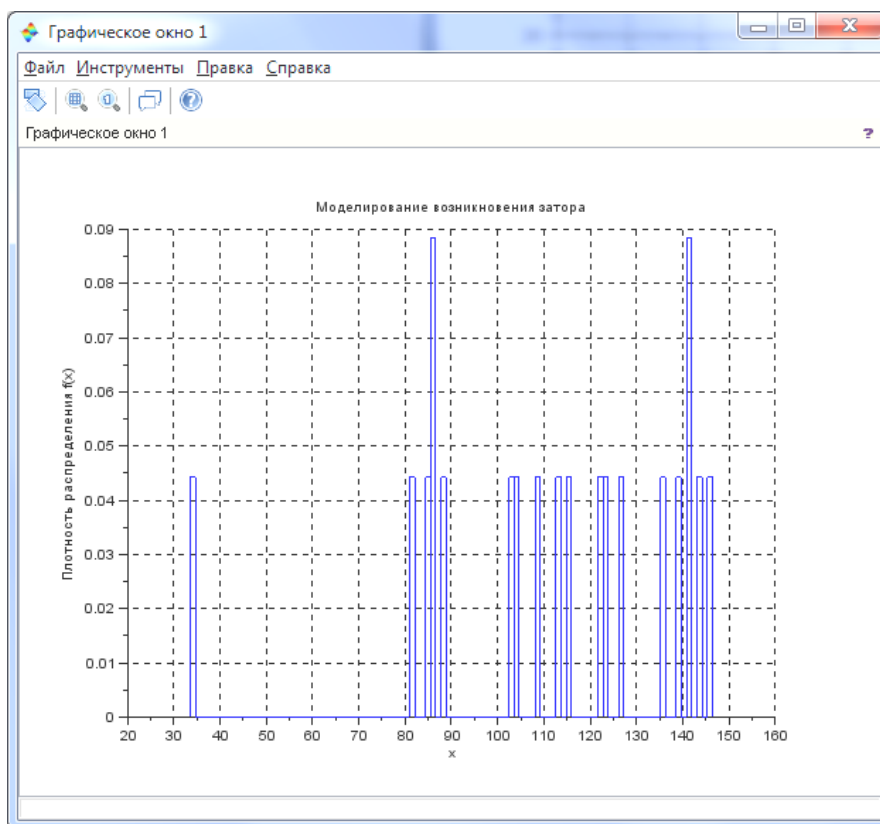


Рисунок 3 – Плотность распределения

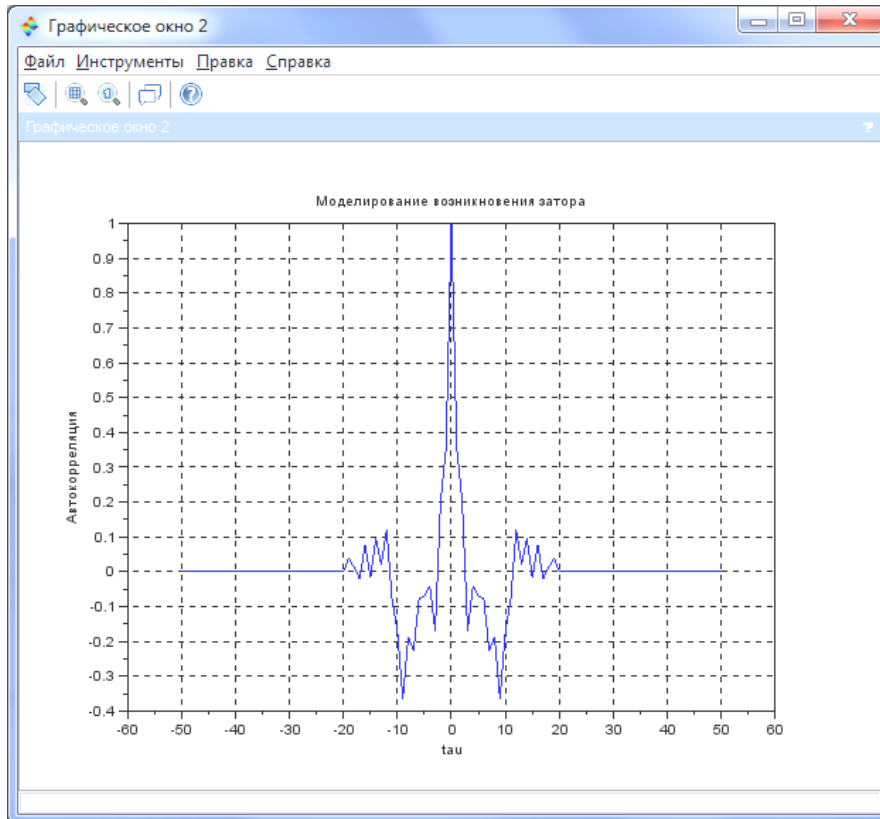


Рисунок 4 – АКФ динамического ряда

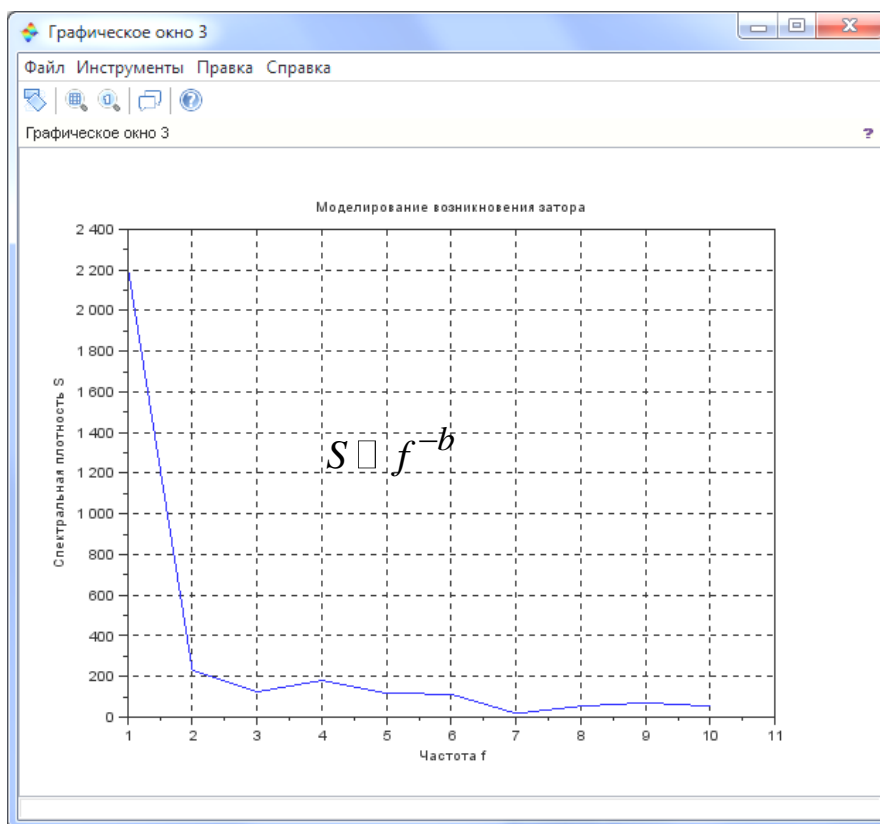


Рисунок 5 – Спектральная плотность

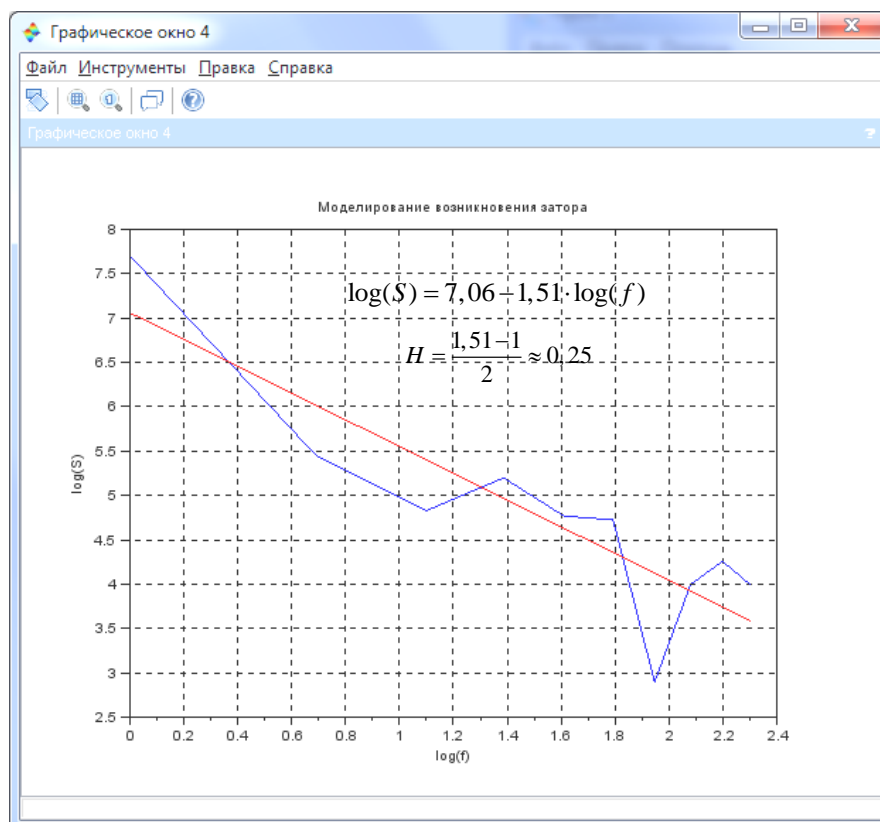


Рисунок 6. – Зависимость $\log(S)$ от $\log(f)$

Выводы

Для транспортных потоков, характерно обнаруженное на практике свойство фрактальности. В связи с этой особенностью актуальной является разработка конструктивных методов анализа возникновения транспортного затора. В данной работе предложена модель транспортного затора с использованием алгоритма Бака-Снеппена. Выполнен анализ плотности распределения, автокорреляционной функции и энергетического спектра, полученного динамического ряда. Определено значение показателя Херста H . Получены следующие результаты: изучаемый временной ряд является антиперсистентным, не подчиняется нормальному распределению, не имеет экспоненциально спадающей АКФ, свойственной случайному ряду. Антиперсистентный розовый шум является характерным свойством транспортного потока. Транспортный поток обладает всеми свойствами изучаемого временного ряда. Поэтому алгоритм Бака-Снеппена можно использовать при моделировании возникновения транспортного затора. Это актуально для повышения эффективности дорожно-транспортной инфраструктуры за счет математического моделирования.

Литература

1. Транспортный поток // <http://www.manytransport.ru/maors-1152-2.html>
2. Бак П. Как работает природа: Теория самоорганизованной критичности. Москва: Либроком, 2013–276 с.
3. Sneppen K., Bak P., Flyvbjerg H., Jensen M.H. Evolution as a self-organized critical phenomenon. // Proceedings of the National Academy of Sciences. 1995. Vol. 92. № 11. P. 5209–5213.
4. Жуков Д.С., Лямин С.К. Генераторы розового шума как инструмент имитационного моделирования. // Ineternum. 2014. № 1. С. 43–46.
5. Модель Бака-Снеппена // http://www.cognitivist.ru/er/kernel/1f_soc.xml

УДК 519.6 (076)

ВЕЛИКАЯ ТЕОРЕМА ФЕРМА И ЕЕ ВЛИЯНИЕ НА РАЗВИТИЕ СОВРЕМЕННОЙ НАУКИ

Потреба Е.Ю.¹, Ефименко К.Н.²

Донецкий национальный технический университет
студент гр. ИС-176¹, кафедра прикладной математики²

E-mail: lampadore@gmail.com

Аннотация:

Потреба Е.Ю., Ефименко К.Н. Великая теорема Ферма и ее влияние на развитие современной науки. Рассмотрена история попыток доказательства Великой теоремы Ферма и ее доказательство профессором математики Э. Уайлсом. Показано влияние Великой теоремы Ферма на современную математику.

Annotation:

Potreba E.Yu., Efimenko K.N. Fermat's Great theorem and its influence on the development of modern science. The history of attempts to prove the Fermat's Great theorem and its proof by the professor of mathematics Andrew John Wiles is considered. The influence of the Fermat's Great theorem on modern mathematics is shown.

Введение

Великая теорема Ферма стала одним из символов труднейшей научной проблемы. Французский юрист и по совместительству великий математик XVII века Пьер Ферма (1601-1665) выдвинул одно любопытное утверждение из области теории чисел, которое впоследствии получило название Великой (или Большой) теоремы Ферма. Это одна из самых известных и феноменальных математических теорем. Наверное, ажиотаж вокруг нее был бы не так силен, если бы в книге Диофанта Александрийского (III век) «Арифметика», которую Ферма частенько штудировал, делая пометки на ее широких полях, и которую любезно сохранил для потомков его сын Сэмюэл, не была обнаружена примерно следующая запись великого математика: «Я располагаю весьма поразительным доказательством, но оно слишком велико, чтобы его можно было разместить на полях». Эта запись явилась причиной последующей грандиозной суматохи вокруг теоремы [1-5].

Знаменитый ученый заявил, что доказал свою теорему. Давайте же зададимся вопросом: действительно ли он ее доказал или банально соврал? Или есть другие версии, объясняющие появление той записи на полях, не дававшей спокойно спать многим математикам следующих поколений?

Целью данной работы является попытка проанализировать состояние вопроса разрешимости Великой теоремы Ферма на современном этапе развития математических наук.

Теорема Ферма и основные попытки ее разрешения

История Великой теоремы увлекательна, как приключение во времени. В 1636 году Ферма заявил, что уравнение вида $X^n + Y^n = Z^n$ не имеет решений в целых числах при показателе степени $n > 2$. Это собственно и есть Большая теорема Ферма. В этой, казалось бы, простой с виду математической формуле Вселенная замаскировала невероятную сложность.

Несколько странным является то, что почему-то теорема опоздала с появлением на свет, поскольку ситуация назрела давно, ведь ее частный случай при $n = 2$ – другая

знаменитая математическая формула – теорема Пифагора, возникла на двадцать два столетия раньше. В отличие от теоремы Ферма, теорема Пифагора имеет бесконечное множество целочисленных решений, например, такие пифагоровы треугольники: (3,4,5), (5,12,13), (7,24,25), (8,15,17) ... (12,16,20) ... (51,140,149) ...

Кто только не пытался доказать теорему Ферма. Любой математик считал своим долгом приложиться к Великой теореме, но доказать ее все никак никому не удавалось.

Сколько бы теорему не проверяли – она всегда оказывалась верна. Был один энергичный программист, который был одержим идеей опровергнуть Великую теорему, пытаясь найти хотя бы одно ее решение методом перебора целых чисел с использованием быстродействующего компьютера. Ни одного решения он, конечно же, не нашел. И никакие компьютеры, хоть даже со сказочным быстродействием, никогда не смогли бы проверить теорему, ведь все переменные этого уравнения (в том числе и показатели степени) могут возрастать до бесконечности.

Самый плодотворный математик XVIII века Леонард Эйлер доказал теорему Ферма для степеней 3 и 4 (вернее, он повторил утерянные доказательства самого Пьера Ферма); его последователь в теории чисел, Лежандр, – для степени 5; Дирихле – для степени 7. Но в общем виде теорема оставалась недоказанной.

В начале XX века (1907) состоятельный немецкий любитель математики по фамилии Вольфскель завещал 100 000 марок тому, кто предъявит полное доказательство теоремы Ферма. Начался ажиотаж. Математические кафедры были завалены тысячами доказательств, но все они, как вы догадываетесь, содержали в себе ошибки. В то время в кругу математиков появилось полупрезрительное прозвище – *ферматист*.

К тому времени появилось доказательство теоремы для показателя степени $n < 100$. Потом для $n < 619$. Надо ли говорить о том, что все доказательства невероятно сложны. Но в общем виде теорема оставалась недоказанной.

Прошло еще 8 лет. Одному прогрессивному английскому профессору математики из Принстонского университета (Нью-Джерси, США) Эндрю Уайлсу показалось, что он нашел доказательство гипотезы Таниямы [2-3]. Прежде, чем предоставить свое доказательство миру, он тщательно проверял его сам, но осознавая, что может иметь субъективную предвзятость, привлекал к проверкам также и других, например, под видом обычных математических заданий он иногда подкидывал смышленным аспирантам различные фрагменты своего доказательства. Позже Уайлс признался, что никто, кроме его жены не знал, что он работает над доказательством Великой теоремы.

И вот после долгих проверок и тягостных раздумий, Уайлс наконец-то набрался храбрости, а может, как ему самому казалось, наглости и 23 июня 1993 года на математической конференции по теории чисел в Кембридже объявил о своем великом достижении. Это, конечно, была настоящая сенсация. Никто не ожидал такой смелости от почти неизвестного математика. Стройные формулы, как штрихи прекрасной картины, предстали перед любопытными взорами собравшихся.

Презентация доказательства, казалось, прошла успешно – ошибок в нем не нашли. Все решили, что произошло-таки масштабное событие: доказана гипотеза Таниямы, а следовательно и Великая теорема Ферма. Но примерно через два месяца, за несколько дней до того, как рукопись доказательства Уайлса должна была пойти в тираж, в ней было обнаружено несоответствие (Кац, коллега Уайлса, заметил, что один фрагмент рассуждений опирался на «систему Эйлера», но то, что соорудил Уайлс, такой системой не являлось), хотя в целом приемы Уайлса были признаны интересными, изящными и новаторскими.

Уайлс проанализировал ситуацию и решил, что проиграл. Можно себе представить, как он всем своим существом почувствовал, что значит «от великого до смешного один шаг». Для него, серьезного ученого-математика, это была трагедия, и он забросил свое доказательство в долгий ящик.

Но вот через год с небольшим, в сентябре 1994 года, во время размышления над доказательством вместе со своим коллегой Тейлором из Оксфорда, последнего неожиданно осенила мысль, что «систему Эйлера» можно заменить на теорию Ивасава (раздел теории чисел). Тогда они попробовали воспользоваться теорией Ивасава, обойдясь без «системы Эйлера», и у них все сошлось. Исправленный вариант доказательства был отдан на проверку и через год было объявлено, что в нем все абсолютно четко, без единой ошибки. Летом 1995 года в одном из первенствующих математических журналов было опубликовано полное доказательство гипотезы Таниямы (следовательно, Великой (Большой) теоремы Ферма), которое заняло весь номер – свыше ста листов.

Таким образом, в конце XX века весь мир признал, что на 360 году своей жизни Великая теорема Ферма, которая на самом деле все это время являлась гипотезой, стала-таки доказанной теоремой. Эндрю Уайлс доказал Великую (Большую) теорему Ферма и вошел в Историю.

За восемь лет труда Уайлс, по существу, свел воедино все достижения теории чисел XX века, выстроив из них одно сверхмощное доказательство. Преследуя свою главную цель, Уайлс попутно создавал совершенно новые доказательства и использовал их в немыслимых ранее сочетаниях с традиционными методами.

С помощью гипотезы Таниямы–Шимуры Уайлс объединил эллиптический и модулярный миры и, тем самым, проложил математике пути ко многим другим доказательствам: проблемы, стоящие в одной области, могут быть решены по аналогии с проблемами из параллельной области. Классические нерешенные проблемы теории эллиптических кривых стало возможным подвергнуть пересмотру, используя все имеющиеся средства и методы теории модулярных форм.

Великая теорема Ферма, несмотря на весь интерес, который она к себе вызывала и продолжает вызывать, все же является проблемой частного характера, то или иное решение которой само по себе вряд ли могло бы значительно расширить горизонт научной мысли в ее современном состоянии.

Тем не менее, доказательство Великой теоремы Ферма можно поставить в один ряд с такими достижениями XX века, как изобретение компьютера, ядерной бомбы и полет в космос (а также радио, телевидения, открытие ДНК...). Хоть о нем и не так широко известно, потому что оно не вторгается в зону наших сиюминутных интересов, как например, телевизор или электрическая лампочка, но оно явилось вспышкой сверхновой звезды, которая, как и все непреложные истины, всегда будет светить человечеству.

Действительно ли Ферма доказал свою теорему [4]?

1. Ферма доказал свою теорему. На вопрос: «Имел ли Ферма точно такое же доказательство своей теоремы?», Эндрю Уайлс заметил: «Ферма не мог располагать таким доказательством. Это доказательство XX века». Мы с вами понимаем, что в XVII веке математика, конечно же, была не та, что в конце XX века – в ту эпоху царица наук еще не обладала теми открытиями (модулярные формы, теоремы Таниямы, Фрея и др.), которые только и позволили доказать Великую теорему Ферма. Конечно, можно предположить, что Ферма догадался другим путем. Эта версия хоть и вероятна, но по оценкам большинства математиков, почти невозможна.

2. Пьеру Ферма показалось, что он доказал свою теорему, но в его доказательстве были ошибки, то есть, сам Ферма был также и первым ферматистом;

3. Ферма свою теорему не доказал, а на полях просто соврал.

Если верна одна из двух последних версий, что наиболее вероятно, то тогда можно сделать простой вывод: великие люди, они хоть и великие, но тоже могут ошибаться или иногда не прочь приврать. Поэтому, читая произведения гениев или слушая их выступления, вы имеете полное право сомневаться в их утверждениях. Прошу заметить, что сомневаться – не значит отвергать.

Выводы

В заключение доклада хочется еще раз отметить очень тесную и глубокую привязку Великой теоремы к реальной геометрии нашего мира. Дело в том, что сам Ферма обладал уникальной способностью очень точного и верного проникновения в математические закономерности наблюдаемых им явлений и свойств. На основе изучения творчества Ферма автору удалось воспроизвести ход его возможных мыслей при формулировании им своей теоремы и дать изображение его метода в виде «фермаскопа» [5], т.е. наглядной геометрической схемы, показывающей неявную (скрытую) структуру числового пространства, несущую в себе не раскрытую до сих пор аксиоматику будущей теории чисел. К сожалению, современники Ферма не смогли ее распознать в весьма коротком его замечании на полях «Арифметики» Диофанта, и развитие математики пошло по другому пути, по пути искусственной формализации числовых пространств, лишь весьма грубо отражающих необыкновенные или мало заметные свойства реального пространства-времени, перед которыми современная наука остановилась в замешательстве, не имея для их изучения адекватного математического аппарата. Но теперь есть надежда, что такой аппарат будет создан. А пока можно исследовать пространство «фермаскопа» традиционными физическими методами и уже сейчас получить интересные данные, используя для этого модельные представления из квантовой теории поля. Так оказывается, что внутреннее пространство квантовых частиц хорошо моделируется пространством «фермаскопа», позволяя увидеть в нем образование и динамику таких полевых волн-частиц, как инстантон и антиинстантон, инвертон и антиинвертон [4]. Кроме того, «фермаскоп» дает наглядное подтверждение минимального кванта времени – темпона, жестко связанного с движением векторов состояний во внутреннем пространстве квантовых объектов. Еще более интересными могут быть применения «фермаскопа» для квантовой интерпретации механики Ньютона и механики теории относительности.

Литература

1. Ивлиев, Ю.А. Великая теорема Ферма и фрактальная геометрия [Текст] / Ю.А. Ивлиев // Успехи современного естествознания. – 2010. – № 11 – С. 101-101.
2. Ферма теорема / Математическая энциклопедия: в 5 т. – Москва : Советская Энциклопедия, 1985. – 5 т.
3. Соловьев, Ю.П. Гипотеза Таниямы и последняя теорема Ферма [Текст] / Ю.П. Соловьев // Соросовский образовательный журнал, т.4, №2. – 1998. – С. 135-138.
4. Стиллвелл, Д. Математика и её история / Д. Стиллвелл. – Москва - Ижевск : Институт компьютерных исследований, 2004. – 530 с.
5. Ивлиев, Ю.А. Реконструкция нативного доказательства Великой теоремы Ферма [Текст] / Ю.А. Ивлиев // Объединенный научный журнал. (Раздел «Математика»). №7. – 2006. – С. 3-9.

УДК 519.6 (076)

ШИФРОВАНИЕ СИМВОЛЬНОЙ ИНФОРМАЦИИ НА МАШИНЕ ТЬЮРИНГА

Шиян А.С.¹, Добровольский Ю.Н.²

Донецкий национальный технический университет
студентка гр. ИС-17а¹, кафедра прикладной математики²

E-mail: an066210@gmail.com

Аннотация:

Шиян А.С., Добровольский Ю.Н. Шифрование символьной информации на машине Тьюринга. Рассмотрены принципы организации и работы машины Тьюринга. Показана реализация двух методов шифрования символьной информации на машине Тьюринга. Приведены примеры программ для машины Тьюринга.

Annotation:

Shiyan A.S., Dobrovolsky Yu.N. Encryption of character information on a Turing machine. The principles of organization and operation of the Turing machine are considered. The implementation of two methods for encrypting symbolic information on a Turing machine is shown. Examples of programs for the Turing machine are given.

Введение

Машина Тьюринга (МТ) – это математическая абстракция, представляющая вычислительную машину общего вида, которая была предложена Аланом Тьюрингом в 1936 г. для формализации понятия алгоритма. МТ состоит из двух частей – ленты и автомата. Лента используется для хранения информации. Она бесконечна в обе стороны и разбита на клетки, которые никак не нумеруются и не именуется. Клетка может быть пустой или содержать один произвольный символ. Содержимое клетки может меняться – в неё можно записать другой символ или стереть находящийся в ней символ [1].

С машиной Тьюринга связаны два конечных алфавита: алфавит входных символов $A = \{a_0, a_1, \dots, a_m\}$ и алфавит состояний $Q = \{q_0, q_1, \dots, q_N\}$. Причем с разными машинами Тьюринга могут быть связаны разные алфавиты A и Q . Состояние q_0 называется пассивным. Считается, что если машина попала в это состояние, то она закончила свою работу. Состояние q_1 называется начальным. Находясь в этом состоянии, машина начинает свою работу.

Входное слово размещается на ленте по одной букве в расположенных подряд ячейках. Слева и справа от входного слова находятся только пустые ячейки (в алфавит A всегда входит пустая буква a_0 – признак того, что ячейка пуста).

Автомат может двигаться вдоль ленты влево или вправо, читать содержимое ячеек и записывать в ячейки символы [2]. Для шифрования символьной информации на машине Тьюринга необходимо написать программу, которая формируется в виде таблицы (табл.1).

Таблица 1. Структура программы для машины Тьюринга

	a_0	a_1	a_2	...	a_m
q_1					
q_2					
...					
q_N					

Слева перечисляются все состояния, в которых может находиться автомат, сверху –

все символы (в том числе и «a₀»), которые автомат может видеть на ленте (какие именно символы и состояния указывать в таблице – определяет автор программы). На пересечениях же (в ячейках таблицы) указываются те такты, которые должен выполнить автомат, когда он находится в соответствующем состоянии и видит на ленте соответствующий символ.

До выполнения программы нужно записать на ленту входное слово, к которому будет применена программа. Входное слово – это конечная последовательность символов, записанных в соседних клетках ленты; внутри входного слова пустых клеток быть не должно, а слева и справа от него должны быть только пустые клетки. Затем установить автомат в состояние «q₁» (указанное в таблице первым) и поместить его под первым символом входного слова.

После этих предварительных действий начинается выполнение программы. В таблице отыскивается ячейка на пересечении первой строки (т. к. автомат изначально находится в состоянии «q₁») и того столбца, который соответствует первому символу входного слова (это необязательно левый столбец таблицы), и выполняется такт, указанный в этой ячейке. На каждом такте автомат может выполнять в строгом порядке только указанные действия:

- 1) записать произвольный символ «a» в доступную клетку;
- 2) перейти в некоторое состояние «q_i»;
- 3) сдвинуться на одну клетку влево (L), вправо (R), или остаться неподвижным (N).

Таким образом, формально действия одного такта можно записать в виде:

$$«a» «q_i» [L \vee R \vee N],$$

где $[L \vee R \vee N]$ – возможность записи любой из букв L, R или N.

Например, такт «K q₂ R» выполняет запись символа «K» в доступную клетку ленты, переход автомата в состояние «q₂», и сдвиг его на одну клетку вправо.

В результате автомат окажется в новой ячейке, соответствующей новому входному символу и новому состоянию автомата, затем выполняется такт из этой ячейки. В процессе работы автомат будет переходить из одной клетки программы (таблицы) в другую, пока не дойдет до клетки, в которой записано, что автомат должен перейти в состояние «q₀». Эти клетки называются клетками останова. При достижении такой клетки, МТ прекращает свою работу.

Цель данной работы – проанализировать возможности машины Тьюринга при реализации двух методов шифрования символьной информации.

Реализация методов шифрования на МТ

В настоящее время существует большое количество различных методов шифрования символьной информации. Для проведения исследования авторами выбраны метод одноалфавитной подстановки и метод перестановки.

Метод одноалфавитной подстановки состоит в следующем. Дан алфавит, состоящий из символов «АБВГД...Я» и некоторое слово. Необходимо зашифровать слово по некоторому ключу, представляющему собой целое число. Каждый символ в слове сдвигается на n позиций влево (n равно целому числу, выбранному в качестве ключа) относительно соответствующего символа в алфавите [3].

Для реализации алгоритма шифрования методом одноалфавитной подстановки при помощи машины Тьюринга необходимо написать программу, позволяющую зашифровать слово по выбранному ключу. На выходе на ленте мы должны получить только зашифрованное слово. Алфавит такого автомата может включать целые числа и буквы. Чтобы использовать двузначные или трехзначные ключи, их можно записывать в системах исчисления, которые позволят поместить ключ в одну клетку на ленте.

Например, зашифруем слово «шифр» по ключу 2. У нас есть алфавит, состоящий из символов «ирфш» (для удобства в качестве алфавита используются только значащие символы шифруемого слова, опять же на принцип работы программы это никак не влияет).

На выходе получим «фрши».

Автомат будет иметь следующие состояния: q_1 – автомат определяет, по какому ключу шифруется слово, и переходит в соответствующее состояние; q_2, q_3 – автомат шифрует слово по ключу 1 или 2. В таблице 2 приведена программа для МТ, реализующая шифрование заданного слова с помощью метода одноалфавитной подстановки.

Таблица 2. Программа для МТ (метода одноалфавитной подстановки)

	1	2	ш	и	ф	р	a_0
q_1	$a_0 q_2 R$	$a_0 q_3 R$					
q_2			$и q_2 R$	$ф q_2 R$	$р q_2 R$	$ш q_2 R$	$a_0 q_0 L$
q_3			$ф q_3 R$	$р q_3 R$	$ш q_3 R$	$и q_2 R$	$a_0 q_0 L$

Алгоритм симметричного шифрования методом перестановки состоит в следующем. Имеется слово, которое необходимо зашифровать по некоторому ключу. Ключ представляет собой последовательность чисел, первое из которых показывает, какой из символов в исходном слове является первым в зашифрованном, второй показывает, какой из символов в исходном слове является вторым в зашифрованном и т. д. Из этого следует, что длина ключа равна количеству символов в слове. Если длина шифруемого слова больше десяти, ключ следует записывать в системе счисления, которая позволяет записать каждый номер символа в одной клетке. Для реализации алгоритма симметричного шифрования методом перестановки при помощи детерминированной машины Тьюринга необходимо зашифровать по заданному несколькими цифрами ключу слово. Работа МТ выглядит при этом следующим образом. Входным словом является ключ, машина должна работать при любом сочетании этих чисел. По завершении работы на ленте должно остаться только зашифрованное слово.

При этом следует учитывать, что на пути у автомата могут встречаться уже напечатанные символы, которые следует пропускать. Зашифруем слово «шифр» по ключу «3421». Получим «фрши». При этом автомат может находиться в следующих состояниях:

- а) q_1 – автомат определяет, какой символ необходимо напечатать, либо прекращает свою работу, если все символы напечатаны (не осталось символов, составляющих ключ);
- б) $q_2 - q_5$ – автомат печатает соответствующий символ;
- в) q_6 – автомат возвращается в начало слова.

В таблице 3 приведена программа, реализующая шифрование заданного слова с помощью алгоритма симметричного шифрования методом перестановки.

Таблица 3. Программа для МТ (алгоритм симметричного шифрования методом перестановки)

	1	2	3	4	ш	и	ф	р	a_0
q_1	$a_0 q_2 R$	$a_0 q_3 R$	$a_0 q_4 R$	$a_0 q_5 R$	$ш q_0 N$	$и q_0 N$	$ф q_0 N$	$р q_0 N$	
q_2	$1 q_2 R$	$2 q_2 R$	$3 q_2 R$	$4 q_2 R$	$ш q_2 R$	$и q_2 R$	$ф q_2 R$	$р q_2 R$	$ш q_6 L$
q_3	$1 q_3 R$	$2 q_3 R$	$3 q_3 R$	$4 q_3 R$	$ш q_3 R$	$и q_3 R$	$ф q_3 R$	$р q_3 R$	$и q_6 L$
q_4	$1 q_4 R$	$2 q_4 R$	$3 q_4 R$	$4 q_4 R$	$ш q_4 R$	$и q_4 R$	$ф q_4 R$	$р q_4 R$	$ф q_6 L$
q_5	$1 q_5 R$	$2 q_5 R$	$3 q_5 R$	$4 q_5 R$	$ш q_5 R$	$и q_5 R$	$ф q_5 R$	$р q_5 R$	$р q_6 L$
q_6	$1 q_6 L$	$2 q_6 L$	$3 q_6 L$	$4 q_6 L$	$ш q_6 L$	$и q_6 L$	$ф q_6 L$	$р q_6 L$	$a_0 q_1 R$

Исходя из приведенных выше программ можно сделать вывод, что реализовать на машине Тьюринга проще алгоритм шифрования методом одноалфавитной подстановки, поскольку конечный автомат будет иметь меньшее количество состояний. Оба алгоритма являются симметричными алгоритмами, т. е. используют для шифровки и дешифровки один ключ данных. В этом состоит их главный недостаток, при получении ключа кто угодно

сможет получить доступ к данным. Но даже если не брать этот факт во внимание, данные алгоритмы не являются криптографически стойкими. Стойким считается алгоритм, атака на который требует от атакующего наличия столь значительных вычислительных ресурсов или огромных затрат времени на расшифровку перехваченных сообщений, что к моменту их расшифровки защищённая информация потеряет свою актуальность [4].

Выводы

Машина Тьюринга согласно тезису Чёрча-Тьюринга, способна имитировать (при наличии соответствующей программы) любую машину, действие которой заключается в переходе от одного дискретного состояния к другому. Принципиальное отличие машины Тьюринга от реальных вычислительных машин состоит в том, что ее запоминающее устройство представляет собой бесконечную ленту. У реальных вычислительных машин запоминающее устройство может быть как угодно большим, но обязательно конечным. Машину Тьюринга нельзя реализовать именно из-за бесконечности ее ленты. В этом смысле она мощнее любой вычислительной машины. Наиболее удобными задачами, решаемыми при помощи машины Тьюринга, являются задачи обработки символьных последовательностей, к которым можно отнести и методы шифрования символьной информации.

Не смотря на то, что машина Тьюринга позволяет в полной мере реализовать простейшие алгоритмы шифрования, нельзя ни отметить, что при использовании большого количества входящих символов, требуется вводить дополнительные состояния автомата, что в свою очередь приводит к увеличению размеров программы.

Литература

1. Пильщиков, В.Н. Машина Тьюринга и алгоритмы Маркова. Решение задач : учебно-методическое пособие / В.Н. Пильщиков, В.Г. Абрамов, А.А. Вылиток и др. – Москва : МГУ, 2006. – 47 с.
2. Машина Тьюринга [Электронный ресурс] / Интернет-ресурс. – Режим доступа : [www/ URL : https://inf1.info/turing](http://www/URL:https://inf1.info/turing). – Загл. с экрана. Дата обращения: 28.03.2019г.
3. Применение машины Тьюринга для реализации алгоритмов шифрования [Электронный ресурс] / Интернет-ресурс. – Режим доступа : [www / URL : https://moluch.ru/conf/tech/archive/88/4317/](http://www/URL:https://moluch.ru/conf/tech/archive/88/4317/) – Загл. с экрана. Дата обращения: 30.03.2019г.
4. Криптографическая стойкость [Электронный ресурс] / Интернет-ресурс. – Режим доступа : [www/ URL : https://ru.wikipedia.org/wiki/](http://www/URL:https://ru.wikipedia.org/wiki/) – Загл. с экрана. Дата обращения: 07.04.2019г.

УДК 621.271.4:004.422.833

ВЫБОР И ОБОСНОВАНИЕ ПРОГРАММНО-МАТЕМАТИЧЕСКОГО АППАРАТА ДЛЯ СОЗДАНИЯ ГЕОМЕТРИЧЕСКОЙ МОДЕЛИ ПОРОДНОГО ОТВАЛА

Е.В.Прокопенко, Юрко Ю.С.

Донецкий национальный технический университет, г. Донецк
кафедра прикладной математики

E-mail: prokopenko1515@rambler.ru

Аннотация:

Е.В.Прокопенко, Юрко Ю.С. Выбор и обоснование программно-математического аппарата для создания геометрической модели породного отвала. В статье рассмотрены вопросы выбора программно-математического аппарата для создания геометрической модели породного отвала. Для рациональных решений данной модели был выбран пакет SURFER.

Annotation:

Prokopenko E.V., Maslo S.V., Grom A.V. Selection and justification of the software and mathematical apparatus for creating a geometric model of the waste dump. The article deals with the choice of software and mathematical apparatus to create a geometric model of the waste dump. For rational decisions of this model, the package SURFER was chosen.

Общая постановка проблемы

Автоматизация геолого-маркшейдерских работ [1,2] неразрывно связана с применением компьютерных технологий. На каждом этапе решения конкретной задачи встаёт вопрос выбора наиболее эффективных средств обработки информации. При этом часто отдаётся предпочтение так называемым "стандартным", широко распространённым пакетам и программам, которые, однако, далеко не всегда являются действительно эффективными.

Постановка задачи. Для решения полномасштабных задач подобного рода используется математическое и программное обеспечение геоинформационных систем типа ArcView, MapInfo, AutoCad, Easy Trace и т. п., для эффективного функционирования которых требуются мощные производительные компьютеры.

Одним из наиболее рациональных решений, является использование пакетов фирмы Golden Software SURFER и GRAPHER для Windows.

Математическую основу пакета SURFER составляют восемь алгоритмов интерполяции двумерных функций в узлах регулярной сетки по исходным данным в произвольных точках области: Кригинг [3], инверсные расстояния, минимизация кривизны, радиальные базовые функции, полиномиальная регрессия, метод Шепарда (комбинация метода инверсных расстояний со сплайнами), триангуляция. Эта база позволяет решать широкий круг задач. Для визуализации применяются два базовых способа: в виде карты изолиний или трехмерной поверхности. Следует отметить не только более простую "технология" работы с графикой, но и то, что возможности работы с изображениями позволяют решать интересные исследовательские и прикладные задачи.

В основе построения изображений в пакете SURFER лежат следующие принципы:

- получение изображения путем наложения нескольких прозрачных и непрозрачных графических слоев;
- импорт готовых изображений, в том числе полученных в других приложениях;

– применение специальных инструментов рисования, а также нанесение текстовой информации и формул для создания новых и редактирования существующих изображений.

В качестве основных элементов изображения используются следующие типы карт: карта изолиний (Contour Map); трехмерное изображение поверхности (3D Surface Map); карта исходных данных (Post Map, Classed Post Map); карта основы (Base Map).

Все типы карт допускается редактировать с помощью встроенных инструментов пакета SURFER, что принципиально важно, так как во многих предметных областях полученная компьютерным способом карта практически служит лишь основой для дальнейшей экспертной оценки. В частности, триангуляция может быть использована для построения трехмерной поверхности по точным значениям данных геодезической съемки, а алгоритм полиномиальной регрессии - для анализа тренда поверхности. Расчет регулярной сетки может выполняться для файлов наборов данных X, Y, Z любого размера, а сама сетка имеет размеры 10000 узлов. Имеется возможность построения поверхности для функций, заданных в явном виде. При расчете поверхности и ее изображения можно определять границу территории произвольной конфигурации.

Для построения пространственной модели породного отвала, достаточно иметь набор точек, которые находятся на поверхности породного отвала с определенными координатами X, Y, Z, которые определяются в результате маркшейдерских съемок породных отвалов. Как известно, все измерения производятся с погрешностями. Для наиболее распространенных графических моделей, построенных на основе линейной интерполяции, можно считать, что минимальная погрешность прогнозирования признака с их помощью составляет, в среднем, примерно 0,8 от точности измерений.

Проверка выполнялась с использованием математической модели, которая представляла собой 5 концентрических окружностей с радиусами соответственно 1,2,..5 см. Были взяты точки на каждой окружности, найдены их высотные отметки на плане, а для каждой точки были найдены координаты X и Y. (Рис.1).

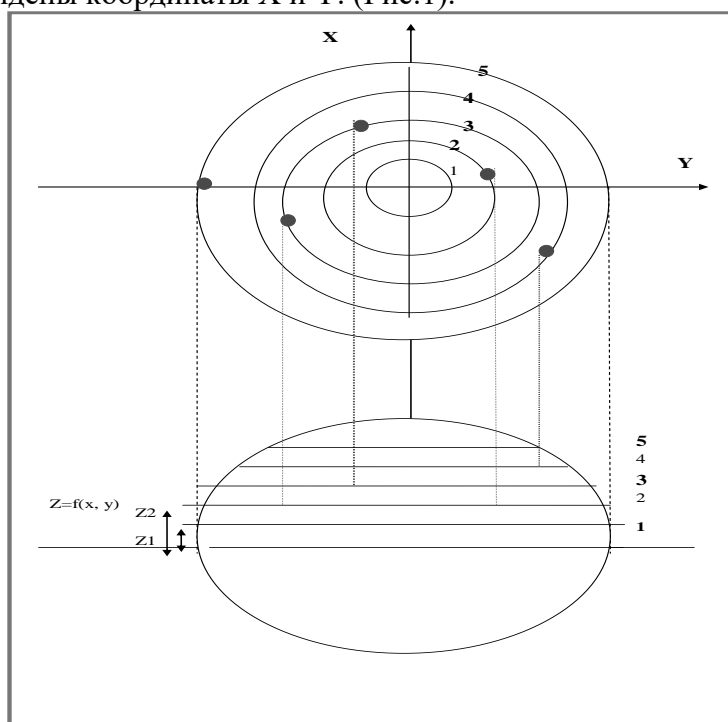


Рис.1.Схема для расчета оценки точности построения изолиний

Для оценки точности выполнено сравнение результатов вычисления с учетом отклонения от тестовой математической модели. Результаты построения данной модели с

учетом различных методов интерполяции показаны ниже. В качестве критерия точности, который численно характеризует условия измерения, принимается средняя квадратическая ошибка, которая определяется по формуле Гаусса:

$$m = \sqrt{\frac{[\Delta\Delta]}{n}} \quad (1)$$

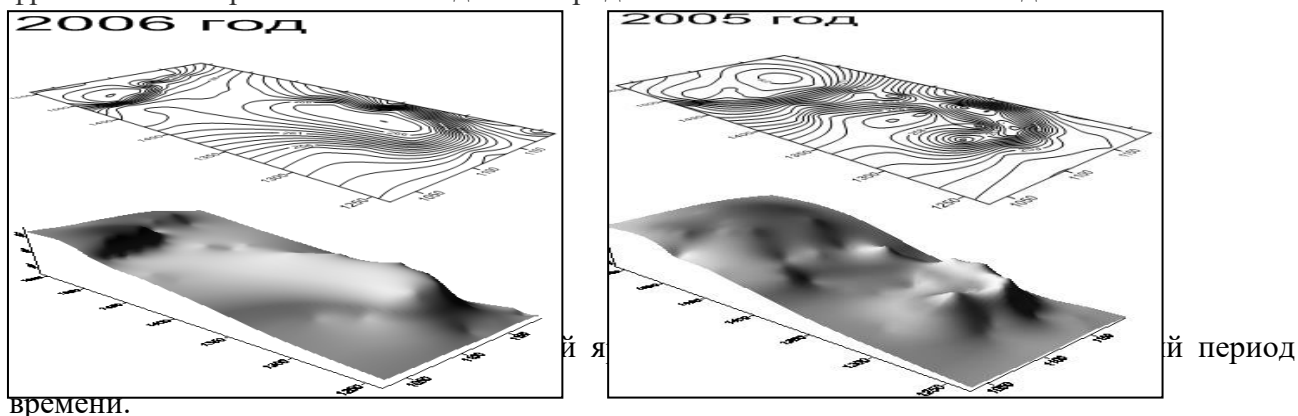
где: m – средняя квадратическая ошибка;

n – количество измерений;

$\Delta\Delta$ – разность между истинным значением измеренной величины и результатом измерения, возведенная в квадрат.

Оценка точности измерений производилась следующим образом: математическая модель, которая использовалась в качестве истинного значения измеренной величины была разбита на квадранты по 90° . Вычисления выполнялись только в первом квадранте. Квадрант был разбит на сегменты по 5° . Таким образом, в расчете были использованы 19 точек.

По результатам маркшейдерской съемки формируется база данных точек замера или точек, полученных в результате тахеометрической съемки породного отвала. Данная база точек содержит координаты X, Y и Z. Для построения трехмерной модели отдельного яруса отвала необходимо выполнить несколько операций. Рассмотрим этапы построения такой модели [4]. Для построения в среде SURFER пространственной модели яруса отвала необходимо занести координаты точек съемки в базу данных SURFER. Далее, по введенным данным строится пространственная модель отдельного яруса. На рисунке 2 показаны фрагменты построения таких моделей породного отвала за 2005...2006 годы.



Используя данные маркшейдерских съемок породного отвала, составляется модель отвала во времени. Вид такой модели показан на рисунке 3.

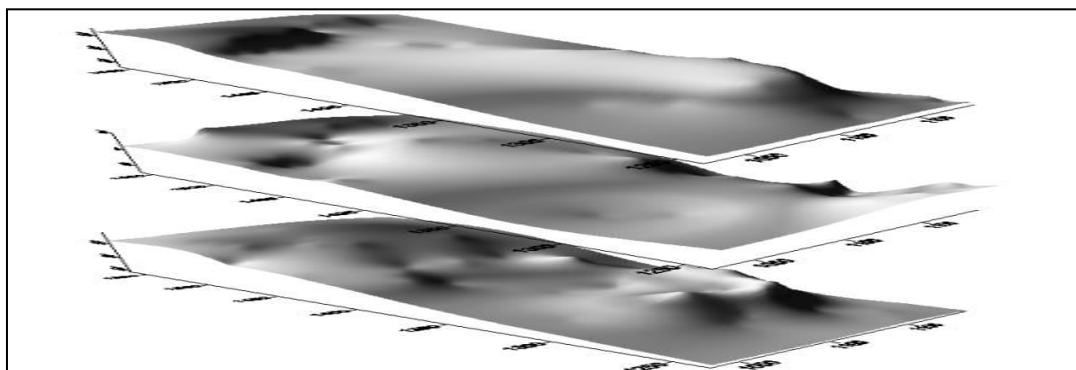


Рис.3.

Поверхности отвала по данным маркшейдерских съемок за определенный период насыпки породы

Выводы.

Таким образом, так как существует несколько вариантов построения поверхности, основанных на использовании методов интерполяции, то на тестовой математической модели, был определен оптимальный метод интерполяции который наиболее точно описывает поверхность породного отвала. Таким образом, в результате построения трехмерной и двумерной модели участков яруса породного отвала за определенный период времени можно получить пространственную модель, которая показывает структуру данного отвала. На основе данной модели можно решить следующие задачи:

1. Установить время отсыпки того или иного участка проходческих работ.
2. Определить состав пород на участках.
3. По составу пород и их свойствам оценить возможные очаги самовозгорания отдельных участков и отвала в целом.
4. Установить участки, на которых сложены породы, пригодные для дальнейшего использования в хозяйственных целях.

Установить участки, на которых возможна разработка с целью доизвлечения полезного компонента (сформировано техногенное месторождение). [5]

Литература

6. М.Н. ДеМерс. Географические информационные системы/ М.Н. ДеМерс. —Пер. с англ. Дата+, 1999. — 288 с.
7. Масюков А.В. Модификации интерполяционного метода Шепарда на основе фундаментальных решений. Вестник ТвГУ-2007г. Серия: Прикладная математика (1[4]). Стр. 99-112. ISSN 1995-0136.
8. Масюков А.В. Итерационный метод интерполяции, основанный на масштабируемом сглаживании/ Масюков А.В., Масюков В.В.— Математическое моделирование, 2005.-т.17.-С.46-56.
9. Прокопенко Е.В. Построение пространственной модели отвала с использованием программного пакета Surfer /С.В.Борщевский, Е.В.Прокопенко// Збірник наукових праць НГУ. — Національний гірничий університет, 2010. —№34, т.1.— С. 82—87.
10. Прокопенко Е.В. Построение пространственной модели участков ярусов породного отвала с использованием программного пакета SURFER//С.В.Борщевский,Е.В.Прокопенко//Ресурсовоспроизводящие, малоотходные и природоохранные технологии освоения недр:13-19 сентября 2010г. — Котону(Бенин), 2010.—С.58—60.

СЕКЦИЯ 2

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

ИУС
МКМ 2019

УДК 004.89, 004.273

ТЕСТИРОВАНИЕ МОБИЛЬНОГО ПРИЛОЖЕНИЯ НА ГРУППЕ ВИРТУАЛЬНЫХ УСТРОЙСТВ

Безсонов И. В., Гудаев О. А.

Донецкий национальный технический университет, г. Донецк
кафедра искусственного интеллекта и системного анализа

E-mail: warilyakeron@gmail.com

Аннотация

Безсонов И.В., Гудаев О.А., Тестирование мобильного приложения на группе виртуальных устройств. Рассмотрена архитектура и программная реализация стенда эмуляции взаимодействия с группой виртуальных мобильных устройств. Показана объектно-ориентированная модель приложения тестирования, использующее ADB-интерфейс ОС Android.

Annotation

Bezsonov I.V., Gudaev O.A., Testing a mobile application on a group of virtual devices. Architecture and software tools for interacting with virtual mobile devices. The object-oriented model of the testing application using the Android OS ADB interface is shown.

Общая постановка задачи

Разработать сервер эмуляции взаимодействия группы виртуальных устройств по интерфейсу ADB . Реализовать выполнение тестовых команд на виртуальных устройствах. Протестировать групповую рассылку команд на множество клиентов. Спроектировать архитектуру тестового стенда. Разработать объектно-ориентированную модель взаимодействия сервера по ADB.

Анализ существующих технологий.

Инструментом, который устанавливается вместе с Android-SDK и позволяет управлять устройством на базе ОС Android называется ADB (Android Debug Bridge - Отладочный мост Android). Работает на всех Android-устройствах, где данный функционал не был намеренно заблокирован производителем. Здесь и далее: PC - ПК, компьютер к которому подключено устройство. ADB - консольное приложение для PC, с помощью которого производится отладка Android устройств, в том числе и эмуляторов. Работает по принципу клиент-сервер. При первом запуске ADB с любой командой создается сервер в виде системной службы (демона), которая будет прослушивать все команды, посылаемые на порт 5037 [3].

ADB позволяет:

- Посмотреть какие устройства подключены и могут работать с ADB.
- Просматривать логи.
- Копировать файлы с/на аппарат.
- Устанавливать/Удалять приложения.
- Удалять (очищать) раздел data.
- Прошивать (перезаписывать) раздел data.
- Осуществлять различные скрипты управления.
- Управлять некоторыми сетевыми параметрами.

Поставляется ADB в составе инструментария разработчика Андроид (Android SDK), который, в свою очередь входит в состав Android Studio. Также для более удобного использования реализованы различные фреймворки, позволяющие ускорить и упростить работу с данным инструментом [2].

Использование adb возможно как с различными физическими android-устройствами, так и с виртуальными устройствами, то есть эмуляторами android в которых разработчики реализовали возможность работы с этим инструментом.

Архитектура тестового стенда

На рисунке 1 предоставлена иллюстрация общего устройства системы, в её состав входят:

- Сервер Adb;
- Физический смартфон на системе android 6.0;
- Виртуальные устройства;
- Сервер ADB.

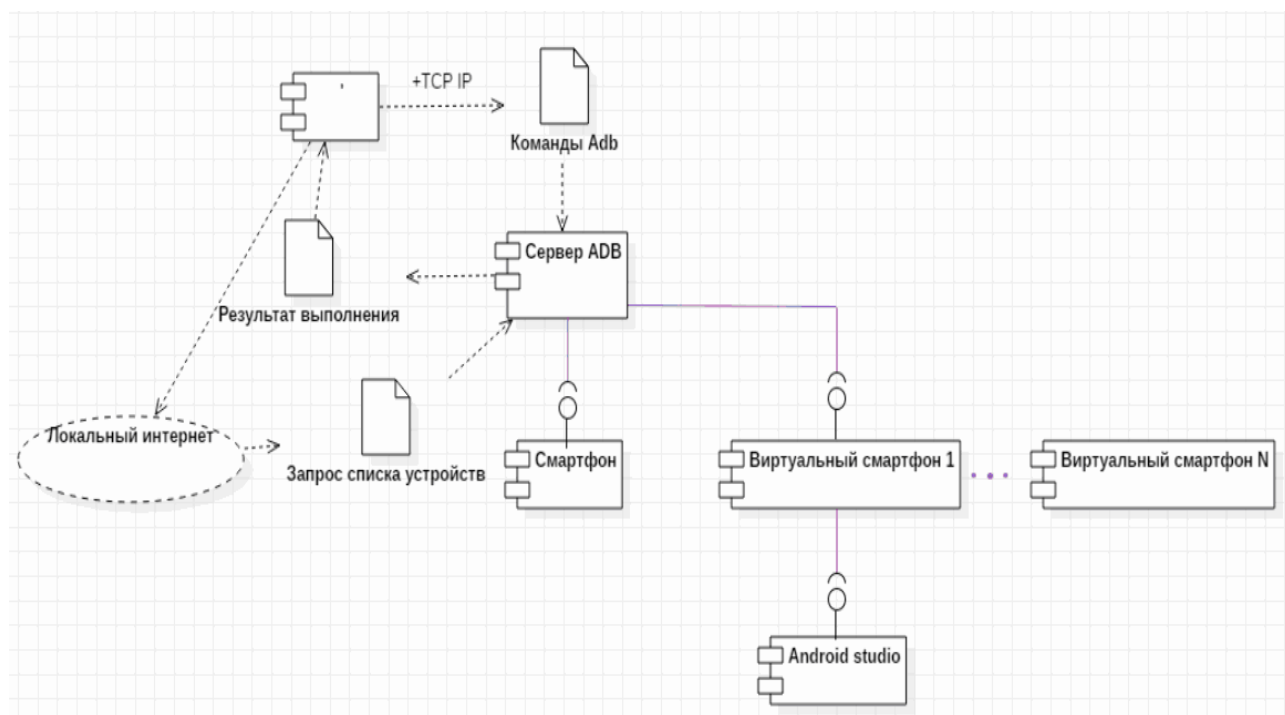


Рисунок 1 - Общая схема компонентов системы

Смартфон и виртуальные устройства соединены с сервером adb посредством локального интернета [2]. Сервер запрашивает список устройств и взаимодействует с ними с помощью adb-команд.

Архитектура клиента

Из диаграммы компонентов (см.рис 2) можно увидеть архитектуру взаимодействия компонентов клиента.

Как видно из диаграммы компонентов (см.рис 2) приложение состоит из классов-видов где каждое окно приложения соответствуют одному классу, класс реализует интерфейс view и обязаны иметь метод Exist который проверяет является ли данный вид верным.

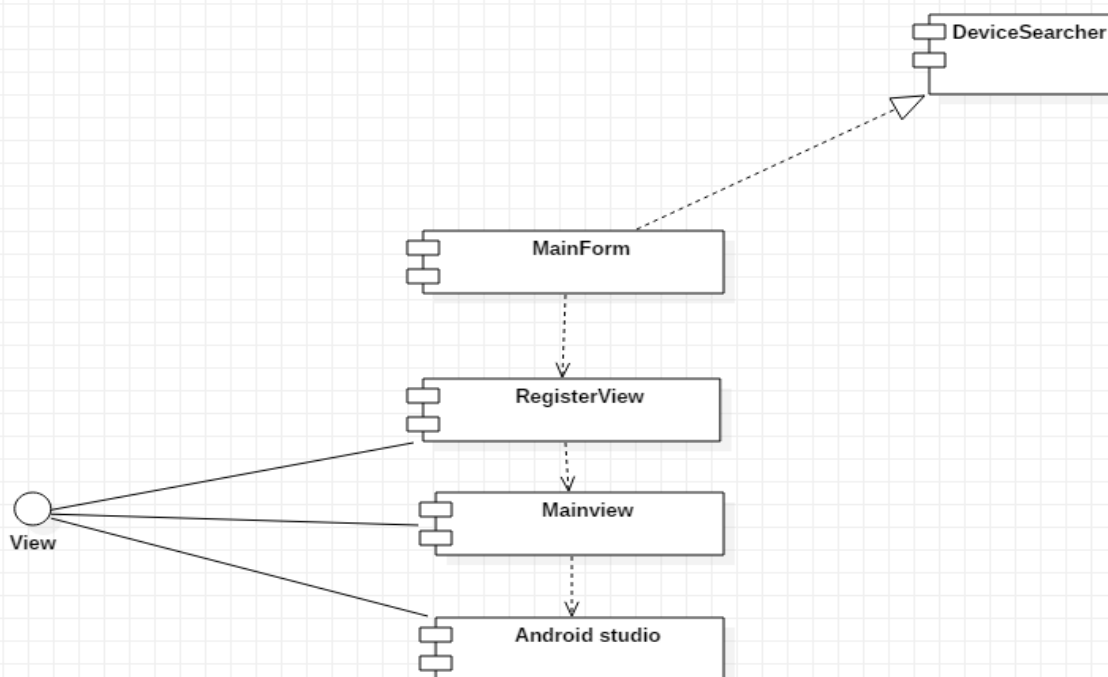


Рисунок 2 - Диаграмма компонентов локальных интернет вещей

Каждый метод вида за исключением последнего возвращает следующий вид для реализации следующего метода. Также приложение содержит класс DeviceFinder который реализует метод поиска устройств для последующего взаимодействия с ними.

Выводы

Разработана объектно-ориентированная модель эмуляции выполнения тестовых команд на группе виртуальных и физических устройств по интерфейсу ADB ОС Android. Получены результаты работы клиентского приложения с группой устройств. Приложение показало высокую эффективность и имеет возможность широкие возможности для улучшения функционала.

Литература

1. Блевотина, А. С. Виртуальные устройства в ООП [Текст] / А. С. Блевотина // учебник «Информационные войны». – Москва: ООО «Патау», 2015. - С. 411-419.
2. Лаврова, А.Н. О дальнейшем развитии словаря смартфонов на Андроиде [Текст] /А. Н. Лаврова, // Сб. ст. Глобальный научный потенциал. – Пенза: ООО «Информационно – технический отдел академии естествознания», 2017. – С. 31 - 33.
3. Кожамкулов, Ж.Ж. Исследование операционных сред для мобильных приложений Андроид [Текст] /Ж. Ж. Кожамкулов, А. И. Наурызбаева // Сб. ст. межд. конф. «Инновационные технологии на транспорте». – Новиград: ООО «Эмсборов», 2018. - С. 27-54.

УДК 004.4

ОБЗОР СУЩЕСТВУЮЩИХ ПРОТОКОЛОВ ДЛЯ ПРОГРАММНОЙ РЕАЛИЗАЦИИ ГОЛОСОВОГО МЕССЕНДЖЕРА

Вязмин В.И., Чернышова А. В.

Донецкий национальный технический университет
кафедра программной инженерии

E-mail: testerreality@gmail.com, chernyshova.alla@rambler.ru

Аннотация:

Вязмин В.И., Чернышова А.В. Обзор существующих протоколов для программной реализации голосового мессенджера. В статье рассмотрены возможные подходы при разработке голосового мессенджера. Выполнен краткий обзор существующих протоколов для передачи аудиоданных в реальном времени.

Annotation:

Viazmin V.I., Chernyshova A.V. Review of existing protocols for software implementation of voice messenger. The article discusses possible approaches in the development of voice messenger. Perform a brief overview of existing protocols for real-time audio communication.

Введение

Голосовые мессенджеры [1] приобрели свою популярность совсем недавно. Развитие данной отрасли берет своё начало в момент, когда средняя скорость интернета по всему миру стала достаточно комфортной для использования данных приложений даже на смартфонах. Мессенджер – это программа, мобильное приложение или веб-сервис для мгновенного обмена сообщениями, в данном случае, голосовыми. Данная отрасль начала развиваться достаточно бурно и эффективно, что привело к созданию по меньшей мере сотен, а то и тысячи мессенджеров.

Мессенджеры – это новый способ коммуницирования между людьми, вне зависимости от географических особенностей, посредством обмена мгновенными сообщениями. В условиях развития интернета как публичного пространства с возможностью доступа к различной персональной информации у пользователей возникает потребность в осознанном потреблении и приватном взаимодействии. Потому и растет актуальность среди приложений, предоставляемых подобные возможности.

Основные подходы разработки голосовых мессенджеров

Каждый мессенджер включает в себя как клиентскую, так и серверную часть. Основной задачей мессенджеров является обеспечение связи (видео, голосовой и т.д.) между пользователями. Взаимодействие между клиентом и сервером, как правило, включает в себя обмен различными сигнальными данными (состояние присутствия, начало звонка и т.д.). Наиболее известным протоколом для этих целей служит XMPP [2]. Важной особенностью при передачи данных (в данном случае аудиоданных) является скорость. Основными протоколами для передачи данных являются TCP [3] и UDP [4]. TCP хоть и является надежным протоколом, однако он не подходит для скоростной передачи потока медиа, а UDP протокол отлично с этим справляется, однако, данный протокол не гарантирует целостность доставленных данных, с чем ему на помощь для решения данной задачи приходит протокол RTP [5]. RTP является основным транспортным протоколом в сетях IP-телефонии [6].

Каждые несколько миллисекунд сервер запрашивает у пользователя его состояние активности (в сети ли он), при хорошем варианте развития событий, при окончании сессии пользователь должен послать сигнал серверу о том, что он вышел из сети, в ином же случае

сервер запросит несколько раз состояние, и если его связь была внезапно прервана, и пользователь об этом не сообщил, соответственно ответа на сигнал об активности не будет, что свидетельствует о том, что пользователь сейчас не в сети (завершил сессию).

Если пользователю необходимо начать сеанс связи с одним из контактов, он посылает запрос об этом на сервер, как правило, за эту часть отвечает SIP-протокол [7]. В соответствии с архитектурой клиент сервер все сообщения подразделяются на запросы и ответы. Запросы передаются от клиента к серверу, ответы – от сервера к клиенту. Для установления соединения вызывающий пользователь формирует SIP-запрос, в котором сообщает серверу адрес вызываемого пользователя, характеристики каналов. В обратном направлении от вызываемого абонента к вызывающему передается ответ на запрос, также с определенными параметрами. Сообщения по протоколу SIP представляют набор текстовых строк. Однако, для этих целей часто используют и другой протокол – Jingle [8].

Jingle – универсальный сигнальный протокол для XMPP. Данный протокол не отвечает за передачу данных, а только за организацию соединения. Два клиента, используя Jingle, оговаривают адреса, порты, тип передаваемых данных, кодеки, тип канала, используемые транспорты и другую информацию, необходимую для установки соединения, по которому уже будет происходить непосредственная передача данных. Сигнальный XMPP протокол Jingle работает поверх только XMPP (без использования SIP) из-за того, что создавать и поддерживать клиенты, реализующие два больших протокола вместо одного достаточно сложно. После согласования по протоколу Jingle начинается RTP/RTCP связь между клиентами, уже без непосредственного участия сервера. На рисунке 1 приведен пример согласования сеанса Jingle.

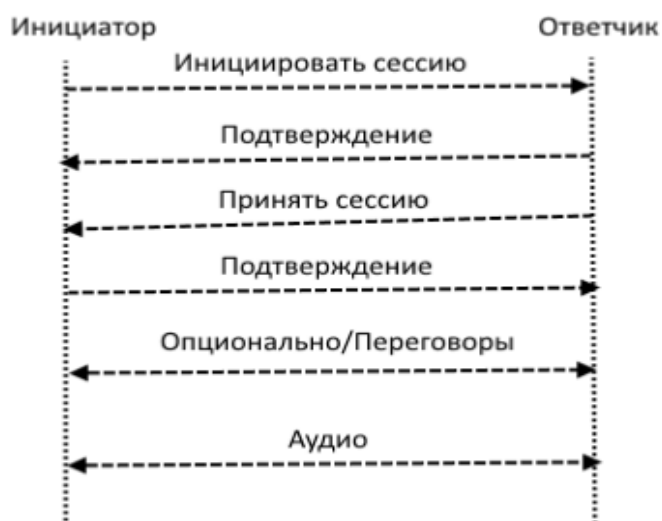


Рисунок 1 – Согласование сеанса Jingle

Стоит заметить, что для правильной работы необходим еще один протокол – SDP [9].

SDP (Session Description Protocol) – протокол согласования типа передаваемых данных (для звука и видео – это кодеки и их форматы, для факсов – скорость передачи и коррекция ошибок) и адреса их назначения (IP и порт). Параметры SDP передаются в теле SIP-пакетов, в данном случае, в теле Jingle-пакетов.

Протокол RTP

RTP (Real-time Transport Protocol) – протокол реального времени, был создан для передачи мультимедийной (ауди, видео), закодированной и упакованной в пакеты, информации по IP сетям в строгих временных рамках. Передача сегментов RTP происходит поверх протокола UDP, соответственно на разных уровнях модели OSI [10]. Использование

протокола UDP, не гарантирующего доставку, связано со строгими временными рамками передачи мультимедийной информации в реальном времени, а также неспособностью протокола TCP работать в режиме реального времени. Поэтому, не смотря на потерю части данных своевременность доставки более важна в данном случае. Как было сказано ранее, данный протокол отвечает за передачу данных между клиентами.

В типичной среде реального времени отправитель генерирует пакеты с постоянной скоростью. Они отправляются через одинаковые интервалы времени, проходят через сеть и принимаются получателем, воспроизводящим данные в реальном времени при их получении. Однако ввиду изменения времени задержки при передаче пакетов по сети, они могут прибывать через нерегулярные интервалы времени. Для компенсации этого эффекта поступающие пакеты буферизуются, придерживаются на некоторое время и затем предоставляются с постоянной скоростью программному обеспечению, генерирующему вывод. Поэтому для функционирования протокола реального времени необходимо, чтобы каждый пакет содержал временную метку – тогда получатель сможет воспроизвести поступающие данные с той же скоростью, что и отправитель.

Реализацию функций RTP контролирует транспортный протокол управления передачей в режиме реального времени RTCP [11] – Real-time Transport Control Protocol. Он также отслеживает качество обслуживания и снабжает соответствующей информацией участников конференции. RTCP использует тот же самый базовый транспортный протокол, что и RTP (обычно UDP), но другой номер порта. RTP работает поверх UDP и может поддерживать передачу данных в реальном времени между несколькими участниками RTP-сеанса

При организации аудио-конференции каждый участник должен иметь адрес и два порта, один для звуковых данных, другой для управляющих RTCP-пакетов. Эти параметры должны быть известны всем участникам конференции. При необходимости соблюдения конфиденциальности информация и пакеты управления могут быть зашифрованы. При аудио конференциях каждый из участников пересылает небольшие закодированные звуковые фрагменты длительностью порядка 20 мсек. Каждый из таких фрагментов помещается в поле данных RTP-пакета, который в свою очередь вкладывается в UDP-дейтаграмму [12].

Польку участники конференции могут как появляться, так и исчезать по своему усмотрению, полезно знать не только, кто из них присутствует в сети в данный момент, но и как до них доходят передаваемые данные. Для этой цели периодически каждый из участников транслирует через порт RTCP мультикастинг-сообщение, содержащее имя участника и диагностические данные. Узел-участник конференции шлет пакет BYE (RTCP), если он покидает сессию. На рисунке 2 представлена структурная схема протокола RTP.

+ Биты	0-1	2	3	4-7	8	9-15	16-31
0	Ver.	P	X	CC	M	PT	Порядковый номер
32	Метка времени						
64	SSRC-идентификатор						
96, если CC>0	[CSRC-идентификаторы]						
96+(CC×32), если X=1	[Заголовок расширения - определенное профилем значение]				[Заголовок расширения - количество блоков данных по 32 бита (EHL)]		
96+ (CC×32)+32	[Заголовок расширения - блоки данных]						
96+(CC×32) +X* (32+32×EHL)	Данные						
если P=1	Заполнение (Padding data)					L	

Рисунок 2 – Структурная схема протокола RTP

В таблице 2 представлено объяснение структурной схемы протокола RTP.

Таблица 2 – Анализ дискриминантных функций

Биты	Значение
0-1	Ver. (2 бита) указывает версию протокола. Текущая версия – 2.
2	P (один бит) используется в случаях, когда RTP-пакет дополняется пустыми байтами на конце.
3	X (один бит) используется для указания расширений протокола, задействованных в пакете.
4-7	CS (4 бита) содержит количество CSRC-идентификаторов, следующих за постоянным заголовком.
8	M (один бит) используется на уровне приложения и определяется профилем. Если это поле установлено, то данные пакета имеют какое-то особое значение для приложения
9-15	PT (7 бит) указывает формат полезной нагрузки и определяет её интерпретацию приложением.
64-95	SSRC указывает источник синхронизации.

Параметр EHL (Extension Header Length) означает количество 32-битных слов в блоке данных расширения заголовка, а L – это последний байт в пакете, определяющий длину области заполнения в байтах (используется для выравнивания в последнем пакете).

Протокол Jingle

Jingle – это дополнение к протоколу XMPP, позволяющее передавать между двумя клиентами аудио- и видеоданные. Согласование джингла может привести к установлению нескольких сеансов RTP (например, один для аудио и один для видео). Приложению следует учитывать, что все сеансы RTP, которые устанавливаются посредством одного и того же согласования Jingle, должны быть синхронизированы для потоковой передачи, воспроизведения, записи и т. д. На рисунке 3 приведен пример протокола Jingle.

```

<iq from='romeo@montague.lit/orchard'
  id='ih28sx61' → id-запроса
  to='juliet@capulet.lit/balcony'
  type='set' → тип запроса
  <jingle xmlns='urn:xmpp:jingle:1'
    action='session-initiate' → сигнатура запроса
    initiator='romeo@montague.lit/orchard'
    sid='e735jvklab7jfee' → id-сессии
    <content creator='initiator' name='voice'>
      <description xmlns='urn:xmpp:jingle:apps:rtp:1' media='audio'>
        <payload-type id='96' name='speex' clockrate='16000'>
        <payload-type id='97' name='speex' clockrate='8000'>
        <payload-type id='18' name='G729'>
        <payload-type id='0' name='PCMU'>
        <payload-type id='103' name='L16' clockrate='16000' channels='2'>
        <payload-type id='98' name='x-ISAC' clockrate='8000'>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:ice-udp:1'
  </iq>
  
```

Кодики которые поддерживае инициатор сессии

Рисунок 3 – Пример протокола Jingle

Выводы

В наше время голосовые мессенджеры приобретают широкую популярность. В связи с этим, становится актуальным вопрос программной реализации подобных мессенджеров. Ознакомившись с протоколами, которые позволяют передавать аудиоданные в реальном времени, можно сделать вывод, что разработка данного мессенджера достаточно трудоемкая работа, которая требует точности. Разнообразие технологий и языков для реализации достаточно обширно, что дает большую свободу в действиях, но основным звеном для создания небольшого десктопного голосового мессенджера все же является связь протоколов XMPP, Jingle и RTP.

Литература

1. Обзор мессенджеров. Лучшие и популярные интернет мессенджеры // VoIPOffice [Электронный ресурс] – Электрон. дан. – 2018. – Режим доступа: <http://www.voipoffice.ru/tags/messendzhery>. – Загл. с экрана.
2. XMPP (Extensible Messaging and Presence Protocol) // Национальная библиотека им. Н. Э. Баумана [Электронный ресурс] – Электрон. дан. – 2017. – Режим доступа: [https://ru.bmstu.wiki/XMPP_\(Extensible_Messaging_and_Presence_Protocol\)](https://ru.bmstu.wiki/XMPP_(Extensible_Messaging_and_Presence_Protocol)). – Загл. с экрана.
3. TCP // Линчакин [Электронный ресурс] – Электрон. дан. – 2019. – Режим доступа: <https://linchakin.com/словарь/t/tcp>. – Загл. с экрана.
4. Транспортные протоколы - UDP // Vanderboot [Электронный ресурс] – Электрон. дан. - 2019. – Режим доступа <http://www.vanderboot.ru/tcp-ip/tcp-udp.php>. – Загл. с экрана.
5. Протокол реального времени RTP // citforum [Электронный ресурс] – Электрон. дан. – 2004. – Режим доступа: http://citforum.ru/nets/semenov/4/44/rtp_4492.shtml. – Загл. с экрана.
6. IP-Телефония: от медных проводов до цифровой обработки криптопримитива / Promwad // habr [Электронный ресурс] – Электрон. дан. - 2013. – Режим доступа: <https://habr.com/ru/company/promwad/blog/188336/> – Загл. с экрана
7. SIP-сообщения // ICX [Электронный ресурс] – Электрон. дан. – 2015. – Режим доступа: <http://www.ixc.ua/109>. – Загл. с экрана.
8. Jingle // jabberworld [Электронный ресурс] – Электрон. дан. - 2013. – Режим доступа: <http://jabberworld.info/Jingle>. – Загл. с экрана.
9. Основы протокола SDP // voipnotes [Электронный ресурс] – Электрон. дан. – 2019. – Режим доступа <https://voipnotes.ru/basic-knowledge-of-sdp/>. – Загл. с экрана.
10. Сетевая модель OSI // Википедия [Электронный ресурс] – Электрон. дан. – 2019. – Режим доступа: https://ru.wikipedia.org/wiki/Сетевая_модель_OSI. – Загл. с экрана.
11. RTCP // Национальная библиотека им. Н. Э. Баумана [Электронный ресурс] – Электрон. дан. – 2017. – Режим доступа: [https://ru.bmstu.wiki/RTCP_\(Real-time_Transport_Control_Protocol\)](https://ru.bmstu.wiki/RTCP_(Real-time_Transport_Control_Protocol)). – Загл. с экрана.
12. Протокол UDP и UDP-дейтаграммы // Компьютерные сети [Электронный ресурс] – Электрон. дан. - 2019. – Режим доступа: <http://iptcp.net/protokol-udp-i-udp-deitagrammy.html>. – Загл. с экрана.

УДК 004.056.5

ПРИМЕНЕНИЕ ЦИФРОВЫХ ВОДЯНЫХ ЗНАКОВ С ЦЕЛЬЮ ЗАЩИТЫ АВТОРСКИХ ПРАВ НА ГРАФИЧЕСКИЕ ИЗОБРАЖЕНИЯ

Шевляков А. И., Ефремченко И. А., Губенко Н. Е.

Донецкий национальный технический университет, г. Донецк
кафедра компьютерного моделирования и дизайна

E-mail: name.ass@gmail.com

Аннотация

Шевляков А.И., Ефремченко И.А., Губенко Н.Е. Применение цифровых водяных знаков с целью защиты авторских прав на графические изображения. В данной статье проведен анализ методов стеганографии, которые позволяют повысить стойкость графических изображений в условиях атак на стегадекодер. Анализируется метод наименее значащих бит (LSB) в комбинации с зашумлением изображения методом белого шума по Гауссу.

Annotation

Shevlyakov A.I., Efremchenko I.A., Gubenko N.E. The use of digital watermarks to protect the copyright of graphic images. This article analyzes the methods of steganography, which allow to increase the durability of graphic images in the face of attacks on a shodecoder. The least significant bit (LSB) method is analyzed in combination with image noise using the Gaussian white noise method.

Постановка проблемы. В настоящее время выросла потребность в приватизации данных и защите авторского права уделяется большое внимание, так как всевозможные мультимедийные данные могут быть использованы в коммерческих целях. Множество пользователей порталов и социальных сетей могут стать жертвами злоумышленников. Всё чаще появляются случаи незаконного использования чужих графических материалов и их владельцам приходится доказывать свое преимущественное право на них. В связи с этим в настоящее время для защиты авторских прав на цифровые изображения все чаще используется технология цифровых водяных знаков, заключающаяся во внедрении в защищаемые графические произведения специальных идентифицирующих правообладателей меток, которые остаются незаметными для человека, но распознаются специализированным программным обеспечением.

Таким образом, проблема защиты авторских прав на цифровые изображения остается актуальной. Решение этой проблемы можно обеспечить с помощью применения модели защиты графических изображений на базе метода внедрения цифровых водяных знаков (ЦВЗ), который позволит усовершенствовать защиту авторских прав на графические изображения в условиях атак на стегадекодер.

Цель статьи – провести анализ методов использования цифровых водяных знаков с целью защиты авторских прав на графические изображения.

Проблема обеспечения подлинности изображений.

Для цифровых изображений, которые находятся в свободном доступе, необходима некая цифровая подпись, с помощью которой автор смог бы контролировать публикации своих работ. В качестве такой информации может выступать текст или цифровое изображение, размещаемое в произвольной области изображения. Цель данного знака – однозначно идентифицировать личность автора. Такие метки были названы цифровыми водяными знаками (ЦВЗ) по аналогии с широко известным способом защиты ценных бумаг

от подделок. Метод защиты графической информации с помощью ЦВЗ является составной частью цифровой стеганографии, науки о незаметном сокрытии одних данных в других.

Цифровые водяные знаки.

В настоящий момент компьютерная стеганография как технология включает в себя несколько областей исследования:

1. Встраивание информации с целью ее скрытой передачи.
2. Встраивание цифровых водяных знаков (ЦВЗ), в частности для защиты авторских прав на электронную продукцию, такую как видео, аудио и графические файлы в электронном виде.
3. Встраивание заголовков.
4. Встраивание идентификационных номеров.

Наибольший интерес вызывают ЦВЗ. Они применяются как для маркирования электронных файлов, так и для встраивания и передачи различной информации по каналам связи [1].

ЦВЗ также могут быть трех видов:

- хрупкие;
- полухрупкие;
- робастные.

Хрупкие ЦВЗ разрушаются при незначительной модификации заполненного контейнера. Они применяются для аутентификации сигналов. Отличие от средств электронной цифровой подписи заключается в том, что хрупкие ЦВЗ все же допускают некоторую модификацию контента. Это важно для защиты мультимедийной информации, так как законный пользователь может, например, пожелать сжать изображение. Другое отличие заключается в том, что хрупкие ЦВЗ должны не только отразить факт модификации контейнера, но также вид и местоположение этого изменения.

Полухрупкие ЦВЗ устойчивы по отношению к одним воздействиям и неустойчивы по отношению к другим. Полухрупкие ЦВЗ специально проектируются так, чтобы быть неустойчивыми по отношению к определенному рода операциям. Например, они могут позволять выполнять сжатие изображения, но запрещать вырезку из него или вставку в него фрагмента.

Под робастностью понимается устойчивость ЦВЗ к различного рода воздействиям на стегоконтейнер. Робастные ЦВЗ могут быть 3-х типов. Это ЦВЗ, которые могут быть обнаружены всеми желающими, хотя бы одной стороной, либо это могут быть ЦВЗ которые трудно модифицировать или извлечь контента (контейнера) [2].

Таким образом, можно выделить актуальную задачу по применению ЦВЗ, которая состоит в защите авторских прав на графические изображения, которые могут быть подвергнуты обрезке краёв, добавлению или удалению каких-либо элементов, сжатию с потерями при сохранении. В ходе поиска решения данной задачи было решено остановиться на открытых стегосистемах, не требующих устойчивости к внешним искажениям. Одним из методов таких систем является метод наименее значащих бит (LSB) в комбинации с зашумлением изображения методом белого шума по Гауссу.

Метод LSB достаточно популярен, это обусловлено его простотой и тем, что он позволяет скрывать в относительно небольших файлах большие объемы информации (пропускная способность создаваемого скрытого канала связи составляет при этом от 12,5 до 30%) [3].

Сокрытие информации в изображении на основе метода наименее значащих бит (LSB) представлен на рис. 1.

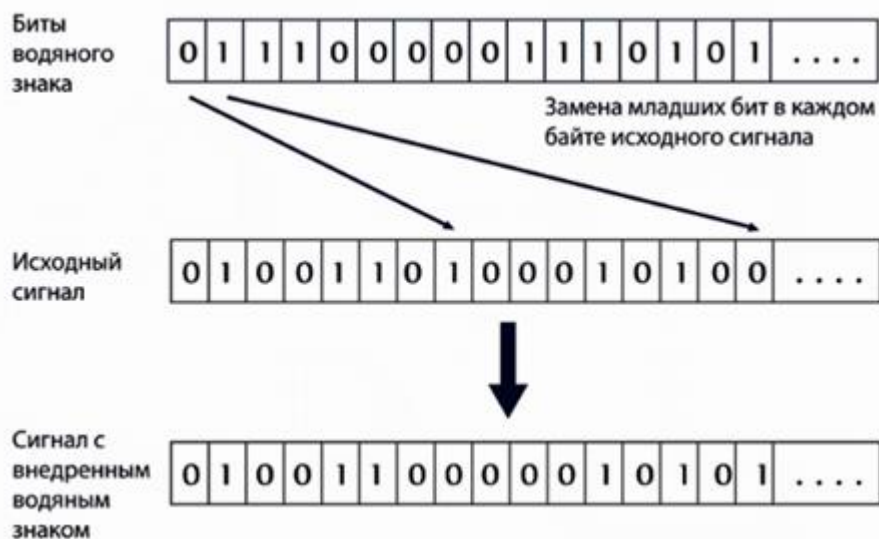


Рисунок 1 - Соккрытие информации в изображении на основе метода наименее значащих бит (LSB) [3]

Метод наименее значащих бит наиболее уязвим к детектированию яркостной статистикой. Аддитивный шум не позволяет с точностью детектировать скрытый водяной знак. Тогда и только тогда, когда мат. ожидание и автокорреляционная функция случайного процесса непрерывного во времени удовлетворяют следующим равенствам:

$$\mu_{\omega}(k) = F\{\omega(k)\} = 0$$

$$R_{\omega\omega}(k_1, k_2) = F\{\omega(k_1)\omega(k_2)\} = \sigma^2\delta(k_1 - k_2)$$

Непрерывный во времени случайный процесс $\omega(k)$, $k \in \mathbb{R}$, является белым шумом (случайным процессом [4]), ω - вектор случайных чисел, является последовательностью отсчетов белого шума, а среднее значение μ_{ω} и матрица автокорреляции $R_{\omega\omega}$ удовлетворяют следующим равенствам:

$$\mu_{\omega} = F\{\omega\} = 0$$

$$R_{\omega\omega} = F\{\omega\omega^T\} = \sigma^2 I$$

Такая автокорреляционная функция имеет данную спектральную плотность мощности:

$$S_{\omega\omega}\{\omega\} = \sigma^2_{\omega}$$

Таким образом, в данном методе важную роль имеют помехи в изображении, так как они играют роль дополнительной защиты (ключа) от детектирования ЦВЗ.

Обоснование выбора наименее значащих бит.

Для определения наименее значащих бит цветного RGB изображения рассмотрим особенности ЗСЧ – зрительной системы человека

К низкоуровневым свойствам ЗСЧ относятся:

а) низкая чувствительность ЗСЧ к малозначимому изменению яркости изображения (3- 5%);

б) низкая чувствительность ЗСЧ к малозначимому изменению контрастности изображения;

в) малое восприятие эффекта маскировки. Сигналы различных источников с различной амплитудой перекрывают друг друга в порядке уменьшения сигнала;

г) низкая чувствительность к малозначимому изменению яркости синего цвета.

Соотношение восприятия $Y = 0,3 R + 0,6 G + 0,1 B$, где R, G, B – каналы яркости единичного пикселя – красный, зеленый и синий цвета соответственно.

Y – полноцветная яркость, которую воспринимает зрительной системой человека. Y зависит от всех каналов цветности, но с различными весовыми коэффициентами.

Исходя из рассмотренной информации, можно сделать вывод о значительно меньшей чувствительности к каналу синего цвета, чем к каналам красного и зелёного.

Рассмотрим алгоритм встраивания ЦВЗ на основе LSB в комбинации с зашумлением изображения методом белого шума по Гауссу:

1. Чтение исходного изображения.
2. Получение числа рядов и столбцов в матрице исходного изображения.
3. Если изображение цветное, извлекается синий канал.
4. Чтение изображения для водяного знака из файла.
5. Получение числа рядов и столбцов в изображении для водяного знака.
6. Если изображение для встраивания цветное, извлекается синий канал.
7. Получение разряда матрицы в котором будет скрыт водяной знак.
8. Выбор бит для записи водяного знака в соответствии с пороговым уровнем яркости.
9. Внедрение цифрового водяного знака в отобранные биты.
10. Добавление шума к изображению с внедрённым водяным знаком.

Однако, в связи с тем, что данная модель основывается на методе LSB, она обладает рядом недостатков:

- отсутствие секретных данных допускает возможность прочтения скрытого сообщения всеми пользователями;
- возможность распознавания наличия сообщения простейшими методами, например, с помощью статистики.

Выводы. Проведенный анализ позволил сделать вывод о том, что для графических изображений с точки зрения защиты авторского права на их файлы принципиально необходимо реализовывать автоматическое подписывание файлов с целью опубликования информации об авторе. Благодаря внедрению ЦВЗ в изображения становится возможным подтверждения и проверка прав разработчика на конкретный мультимедийный файл. Цифровые водяные знаки также являются эффективной защитной мерой для соблюдения прав интеллектуальной собственности, их использование позволяет защитить автора от таких правонарушений, например, как подмена авторства и отказ от авторства.

Данная модель является эффективной благодаря комбинации с зашумлением изображения методом белого шума по Гауссу, так как:

- аддитивный шум не позволяет с точностью детектировать скрытый водяной знак;
- помехи в изображении играют роль дополнительной защиты (ключа) от детектирования ЦВЗ

В связи с этим, целесообразно использовать рассмотренный алгоритм встраивания ЦВЗ на основе LSB в комбинации с зашумлением изображения методом белого шума по Гауссу с целью защиты авторских прав на графические изображения.

Литература

1. Введение в цифровую стеганографию [электронный ресурс] // Портал магистров ДонНТУ: [сайт]. [2006]. URL: <http://masters.donntu.org/2006/fvti/khotov/library/stego.htm>
2. Цифровые водяные знаки [электронный ресурс] // Образовательная социальная сеть KazEdu.kz: [сайт]. [2009-2017]. URL: <https://www.kazedu.kz/referat/133581/2>
3. Замена наименее значащего бита или LSB [электронный ресурс] // Nestego.ru: [сайт]. URL: <http://www.nestego.ru/2012/07/lbs.html>.
4. Яковлев В.А. Защита информации на основе кодового зашумления. Часть 1. Теория кодового зашумления. / Под ред. В.И. Коржика.– С.Пб.: ВАС, 1993.–245с.

УДК 004.42

МУЗЫКАЛЬНЫЙ ПЛЕЕР С ФУНКЦИЕЙ СИНХРОНИЗАЦИИ РИТМА МУЗЫКИ С РИТМОМ ШАГА ЧЕЛОВЕКА

Комаричев Р. Е., Чернышова А. В.

Донецкий национальный технический университет
кафедра программной инженерии

E-mail: komarichev_pi15b@mail.ru, chernyshova.alla@rambler.ru

Аннотация:

Комаричев Р. Е., Чернышова А. В. Музыкальный плеер с функцией синхронизации ритма музыки с ритмом шага человека. Рассмотрены существующие аудиоплееры для мобильных устройств. Определены требования к разрабатываемому приложению. Спроектирован макет пользовательского интерфейса. Описан алгоритм определения моментов ударов в музыке.

Annotation:

Komarichev R. E., Chernyshova A. V. Music player with function of music rhythm and person's step rhythm synchronization. Reviewed existing audio players for mobile devices. Defined requirements for the developed application. Designed layout of user interface. Described the algorithm for detection of beats in music.

Общая постановка задачи

В ходе большого количества экспериментов было подтверждено благоприятное воздействие музыки на здоровье человека: она может укреплять иммунную систему, приводить к снижению заболеваемости, улучшать метаболизм, заряжать энергией или успокаивать [1].

Одна из основных составляющих музыки — это темп. В некоторой доле от него зависит настроение, которое будет передавать композиция. Основываясь на BPM (англ. beats per minute — частота ударов в минуту), композицию можно отнести к спокойной или энергичной. Соответственно, звучание будет задавать настрой для отдыха, либо, к примеру, для занятий спортом.

Целью данной работы является создание плеера для мобильных устройств, который мог бы не просто воспроизводить музыку, а имел бы функцию синхронизации её ритма с ритмами человека, в частности с ритмом шага. Имея такое приложение на своём смартфоне, человек мог бы двигаться в комфортном для него темпе, а подходящее музыкальное сопровождение подбиралось бы автоматически из всей его аудиотеки — собрался он на пробежку, либо же медленно прогуливается по пейзажному парку.

Обзор существующих плееров для мобильных устройств

Среди существующих приложений из данного сегмента явно прослеживаются общие черты, хотя некоторые представители всё же выделяются некоторыми особенностями. Вот несколько распространённых музыкальных плееров для Android:

– AIMP — может воспроизводить все популярные и не только форматы: асх, mp3, wav, ogg, aac, flac, mp4, m4a и др., имеет поддержку интернет-радио и плейлистов сасе в ненарезанных flac. В нём присутствует хороший эквалайзер с пресетами. Во время проигрывания музыки плеером можно управлять с помощью виджета на рабочем столе и на экране блокировки. Пользователь может выбрать светлую или тёмную тему интерфейса. Навигация осуществляется через боковое меню;

– BlackPlayer — имеет приятный интерфейс в чёрной цветовой схеме с красивыми анимациями. Поддерживает стандартные звуковые файлы, такие как flac, mp3, ogg, wav. Эквалайзер уступает AIMP'у, виджет для рабочего стола менее содержателен, однако в плюсы стоит записать возможность скробблинга (персонализация своих музыкальных предпочтений при прослушивании путём создания своего музыкального профиля с помощью специального приложения-скробблера) на Last.fm и получение данных о композициях и коллективах, включая изображения, истории, тексты, оттуда же. Навигация осуществляется свайпами влево и вправо;

– Pi Music Player — как и предыдущие плееры, имеет эквалайзер и виджет на рабочем столе и экране блокировки. Навигация сопровождается плавными переходами и анимациями. Особыми являются функции обрезки рингтонов и обмена музыкой с другими людьми;

– PlayerPro — не смотря на распространение материального дизайна, остаётся верен старому стилю с выпуклыми иконками и глянцевыми цветами. Простой интерфейс с боковой панелью позволяет быстро перемещаться по нужным экранам. В отличие от многих других плееров, в PlayerPro можно редактировать меню. Для составления пользовательских плейлистов и миксов предусмотрена специальная система рейтинга, где каждый трек можно оценить. Рейтинг учитывается и в отдельном меню представлены все треки с отмеченными оценками. В PlayerPro есть достаточно простой эквалайзер, но зато в настройках можно найти много чего полезного: скробблинг, подгрузка текстов песен, настройка оформления с темами и др.

Функциональные требования к плееру

Проанализировав существующие приложения, можно выделить список требований, которые обязательно должны быть реализованы в первую очередь:

- воспроизведение звуковых файлов в формате mp3;
- создание пользовательских плейлистов;
- воспроизведение файлов в прямом, случайном или пользовательском порядке (здесь «пользовательский порядок» — определение пользователем одноразовой очереди воспроизведения без создания нового плейлиста);
- группировка композиций по альбому, исполнителю, папкам;
- регулирование громкости;
- регулирование уровней частот с помощью эквалайзера;
- отображение подробной информации о композициях;
- отображение подсказок и информационных сообщений, а также сообщений об ошибках при необходимости.

Также в список обязательных стоит включить главные особенности, выделяющие разрабатываемое приложение на фоне остальных:

- определение BPM композиции и нахождение моментов ударов;
- синхронизация ударов композиции с шагом пользователя.

На рисунке 1 представлена диаграмма прецедентов, изображающая взаимодействие пользователя с приложением.

Нефункциональные требования к плееру

Основное достоинство хорошего пользовательского интерфейса заключается в том, что человек всегда чувствует, что он управляет программным обеспечением, а не программное обеспечение управляет им.



Рисунок 1 — Диаграмма прецедентов

Для создания у пользователя такого ощущения «внутренней свободы» интерфейс должен обладать рядом свойств:

- естественность;
- согласованность;
- дружелюбность;
- принцип обратной связи;
- простота;
- гибкость;
- эстетическая привлекательность [2].

Учитывая перечисленное, пользовательский интерфейс должен удовлетворять требованиям:

- на экране не должно быть лишней информации и минимум необязательной;
- приложение должно корректно отображаться на экранах всех размеров с разной плотностью пикселей;
- для каждой страницы необходимо предусмотреть как книжную, так и альбомную ориентацию;
- цветовая палитра должна состоять не более, чем из трёх цветов. Допускается присутствие нескольких дополнительных оттенков основных цветов;
- в оформлении не должна использоваться сложная графика. Приветствуется материальный дизайн;
- должна быть создана отдельная страница для текущей проигрываемой композиции, где будет отображена информация о ней, обложка альбома и панель управления

воспроизведением (временная полоса, кнопки переключения композиций, регулятор громкости);

- должна быть создана отдельная страница для управления плейлистами.

На рисунке 2 изображены макеты страниц, описанных в двух последних требованиях.



Рисунок 2 — Макеты книжных и альбомных вариантов страниц воспроизведения (вверху) и управления плейлистами (внизу)

Системные требования

- приложение должно работать на мобильных устройствах под управлением ОС Android версии 5.0 и выше (уровень API 21 и выше);
- приложение должно занимать не более 256 Мб памяти хранилища.

Требования к безопасности

- приложению необходим доступ к сенсорным датчикам устройства для возможности распознавания шагов человека;

- приложению требуются особые разрешения READ_EXTERNAL_STORAGE и WRITE_EXTERNAL_STORAGE для доступа к аудиофайлам на устройстве;
- приложению требуется особое разрешение FOREGROUND_SERVICE для работы в фоновом режиме.

Определение моментов ударов

Человек на слух определяет ритм музыки, распознавая периодические последовательности ударов. Удар можно заметить по контрасту громкости в короткий промежуток времени. Другими словами, в момент резкого скачка силы звука. С другой стороны, если человек будет слушать громкий, но продолжительный монотонный сигнал, ни одного удара он там не услышит. Однако, чувство ритма — это врождённая черта человека. Необходимо научить распознавать ритм компьютер?

Один из методов называется «поиск пиков звуковой энергии». В данном методе определяем изменения энергии звука, вычисляя мгновенную энергию звука и сравнивая её со средней энергией. Допустим, работаем в режиме стерео и имеем два набора значений A_L и A_R — значения амплитуды звука, записанные через равные промежутки времени для левого и правого канала соответственно. Мгновенной энергией назовём суммарную энергию с обоих каналов за короткий промежуток времени. К примеру, имея частоту дискретизации сигнала, равную 44100 Гц, для вычисления можем брать окна по 1024 значения с каждого канала — это покроет примерно 0,025 секунды. Средняя энергия должна вычисляться не по всей композиции, а в пределах небольшого промежутка времени вокруг текущей мгновенной энергии, т.к. в одной композиции могут быть и быстрые партии в начале и медленные в конце, и их усреднённое значение в итоге будет малополезным. Возьмём буфер длительностью около секунды — 44032 значения — и обозначим его B_L для левого канала и B_R для правого.

Итак, для каждого набора из 1024 значений выполняем следующие действия:

1. Для значений из A_L и A_R найдём мгновенную энергию e по формуле:

$$e = e_{stereo} = e_{left} + e_{right} = \sum_{i=i_0}^{i_0+1024} (A_{L_i}^2 + A_{R_i}^2) \quad (1)$$

2. Вычислим локальную среднюю энергию E для буферов B_L и B_R по формуле:

$$E = \frac{1024}{44032} \cdot \sum_{i=0}^{44032} (B_{L_i}^2 + B_{R_i}^2) \quad (2)$$

3. Сравним мгновенную энергию e с локальной средней энергией E , умноженной на коэффициент чувствительности C . Если e превосходит $C \cdot E$, тогда можно сказать, что обнаружен удар.

Это упрощённая версия алгоритма, скорость и точность которого можно повысить. Учитывая тот факт, что в вычислениях используются не отдельные значения амплитуды, а суммы их наборов по 1024 в каждом, нет необходимости хранить каждый экземпляр. Для экономии сил заменим каждый такой набор единственным числом в соответствии с формулой (1) и обозначим список сумм как I . Тогда формула (2) превратится в следующую:

$$E = \frac{1}{43} \cdot \sum_{i=0}^{43} I_i \quad (3)$$

Явный недостаток этого алгоритма в том, что коэффициент чувствительности C — один для всех. В одних музыкальных жанрах удары будут чётко различимы, в других же могут возникнуть трудности при их выделении среди прочих шумов. Есть способ найти оптимальный коэффициент через дисперсию энергий (4). Чем меньше разброс значений в буфере, тем выше должна быть чувствительность алгоритма.

$$D = \frac{1}{43} \cdot \sum_{i=0}^{43} (U_i - E)^2 \quad (4)$$

В формуле (5) подберём такие коэффициенты, чтобы при $D=200$ значение C было равно 1.0, а при $D=25$ — $C=1.45$.

$$C = -0.0025714D + 1.5142857 \quad (5)$$

На данном этапе алгоритм позволяет довольно точно определять моменты ударов, если они не сильно зашумлены. В остальных случаях результаты весьма приблизительные [3].

Выводы

На основании анализа существующих приложений составлен список требований к разрабатываемому продукту. Главная особенность, которая будет выделять его среди аналогов, — это функция подбора композиций в такт шагов человека. Для достижения этой цели предлагается использовать алгоритм «поиска пиков звуковой энергии». Описанный алгоритм является неплохим началом, однако для пользования этого недостаточно. На самом деле довольно малая часть композиций будет иметь чётко выраженные удары, гораздо чаще он не будет справляться со своей задачей. Это можно исправить, снизив количество шума, например, предварительно выделив только частоты, в диапазоне которых обычно звучат инструменты, задающие ритм (ударные, бас-гитара и др.).

Литература

1. Серебрякова, Е. А. Влияние музыки на психофизическое состояние человека / Е. А. Серебрякова // Вестник БГУ. — 2014. — №1. — С.258-262.
2. Гультияев, А. К. Help. Разработка справочных систем: учебный курс / А. К. Гультияев. — СПб: Питер, 2004. — 270с.: ил.
3. Patin F. Beat detection algorithms [Электронный ресурс] flipcode. — URL: <http://www.flipcode.com/misc/BeatDetectionAlgorithms.pdf> (дата обращения: 27.04.2019).

УДК 004.4

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ АУДИО СТЕГАНОГРАФИИ

Толбатова А. С., Чернышова А. В.

Донецкий национальный технический университет, г. Донецк
кафедра программной инженерии

E-mail: anntolb0789@gmail.com, chernyshova.alla@rambler.ru

Аннотация:

Толбатова А.С., Чернышова А.В. Проектирование и разработка приложения для аудио стеганографии. В статье рассматривается общая схема стеганографического процесса, приведены общие требования к разрабатываемой системе *Sound_stego*, а также описан интерфейс приложения и его функциональные возможности.

Ключевые слова: стеганография, аудио, wav, mp3, интерфейс, сокрытие информации.

Annotation:

Tolbatova A.S., Chernyshova A.V. Design and development of applications for audio steganography. The article discusses the general scheme of the steganographic process, describes the general requirements for the developed *Sound_stego* system, and also describes the interface of the application and its functionality.

Keywords: steganography, audio, wav, mp3, interface, information hiding.

Введение

Сейчас практически невозможно представить нашу жизнь без интернета. Более 3 млрд. человек пользуются всемирной паутиной. Так как данная технология развивается очень быстро, возникает ряд проблем. Наиболее актуальной из этого ряда является обеспечение безопасности самой сети. Каждый день пользователи обмениваются данными, которые должны оставаться засекреченными. Интернет должен обеспечить защитой передаваемые данные и самого пользователя от всякого рода атак и угроз.

Важным сегментом информационной безопасности является стеганография. Стеганография – наука о сокрытии передачи информации. Её задачей является скрытие самого факта передачи данных [1].

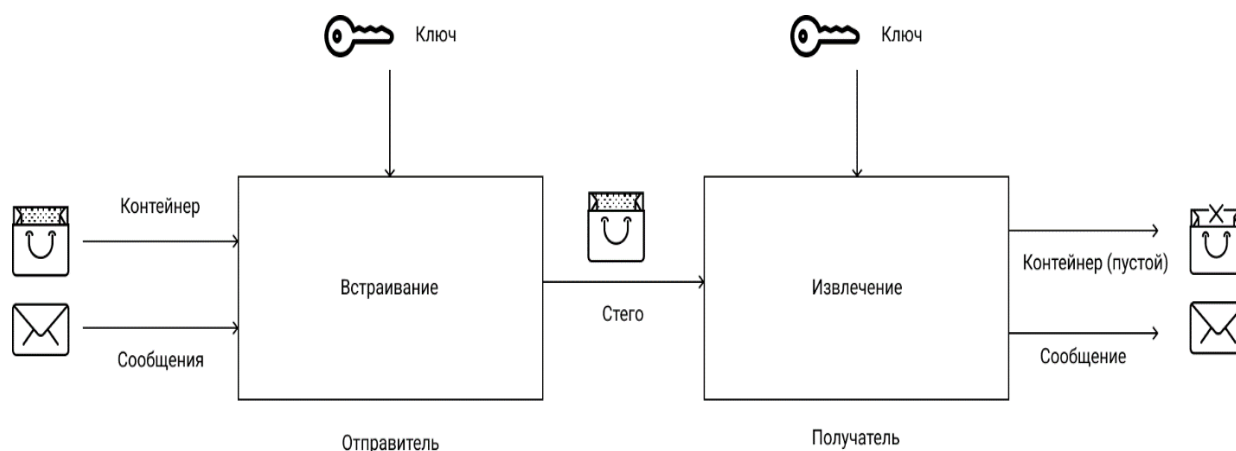


Рисунок 1 – Общая схема стеганографического процесса

В качестве контейнера могут использоваться файлы различных форматов. Цифровая аудио стеганография предполагает использование в качестве контейнера аудио файла.

Обзор программного обеспечения для аудио стеганографии

В 1999 году Флориан Хайденрик разработал программу Mp3tag. Это бесплатный редактор метаданных для большинства аудио форматов с закрытым кодом.

Mp3tag представляет собой довольно простой в использовании программный продукт, с помощью которого можно быстро редактировать теги в разных аудио файлах. Можно вносить изменения в ID3-теги, в комментарии MP3 и Ogg файлов, а также можно производить замену специальных символов или слов. Из всех возможностей программы можно отметить функции импорта/экспорта информации, поддержку Unicode и работу с онлайн-базой данных freedb, а также осуществление группового переименования файлов на основе информации в тегах [2, 3].

Утилита поддерживает такие аудио форматы: ALAC, AAC, FLAC, APE, MP3, MP4, MPEG-4, MPC, OGG, Optim FROG OFR, OFS, SPX, TAK, TTA, WMA, WV.

Разработка требований к приложению для стеганографии

Разрабатываемая программная система будет служить для сокрытия информации в аудио файлах. Её интерфейс должен быть простым и понятным в использовании, чтобы пользователь ясно и чётко понимал для чего необходимо каждое действие и применял их правильно [4].

Общие требования к системе

Данный программный продукт должен соответствовать таким требованиям:

- программная система должна работать на Windows 7/8/8.1/10;
- программная система должна проверять наличие необходимых библиотек для работы;
- программная система должна минимизировать количество информации, вводимой пользователем, либо максимально упростить этот процесс;
- программная система должна обрабатывать исключительные ситуации, возникающие в ходе работы системы, для избегания прерываний в работе программной системы;
- программная система должна позволять пользователю загружать исходный аудио файл и сохранять его после применения стеганографии;
- программная система должна встраивать данные в аудио файл так, чтобы не приводить к значительным искажениям аудио.

Системные и программные требования

Программная система должна корректно функционировать на компьютерах, отвечающих следующим требованиям:

- процессор Pentium-1000 с тактовой частотой, ГГц – 1,1, не менее [5];
- материнскую плату с FSB, ГГц - 5, не менее;
- оперативную память объёмом, Гб - 3, не менее.

А также программное обеспечение компьютеров должно отвечать наличию следующих программных библиотек:

- .NETframework 4.0.

Требования к пользовательскому интерфейсу

Интерфейс программной системы должен следовать ориентированному на пользователя проектированию (user-centered design), в котором потребностям, желаниям и

ограничениям конечных пользователей уделяется обширное внимание на каждой стадии процесса проектирования.

Основное преимущество хорошего пользовательского интерфейса заключается в том, чтобы пользователь чувствовал, что именно он управляет программным обеспечением, а не наоборот. Для этого он должен удовлетворять следующим характеристикам:

- естественность;
- согласованность;
- дружелюбность;
- принцип «обратной связи»;
- простота;
- эстетическая привлекательность.

Естественность заключается в том, чтобы не заставлять пользователя менять привычные для него способы решения задачи, а также использовать знакомые для пользователя понятия и метафоры.

Согласованный интерфейс обеспечивается одинаковой реакцией на семантически одинаковые команды в рамках всего продукта.

Дружелюбный интерфейс ведёт себя таким образом, чтобы избежать заранее некорректных ситуаций и предупреждает пользователя об этом.

Принцип обратной связи заключается в том, чтобы показывать пользователю процесс выполнения долгих задач, а также возможность их отмены.

Простота интерфейса должна обеспечиваться одновременно и лёгкостью в его изучении и предоставлении доступа ко всей функциональности программы на данном этапе.

Эстетическая привлекательность обеспечивается тем, чтобы пользователь не только понимал предоставленную информацию, но и мог бы сосредоточиться на её важных аспектах [6].

Исходя из вышеупомянутых свойств интерфейса, пользовательский интерфейс программной системы должен удовлетворять следующим требованиям:

- необходимо наличие двух кнопок загрузки и сохранения, соответственно, исходного и модифицированного аудио файлов;
- необходимо наличие кнопки и поля для загрузки встраиваемой информации;
- необходим вариант выбора формата аудио файла;
- необходима кнопка для запуска стеганографии;
- необходим статус-бар для контроля работы приложения;
- необходимы уведомления для пользователя о произошедших ошибках.

Макет интерфейса программы представлен на рисунке 2.

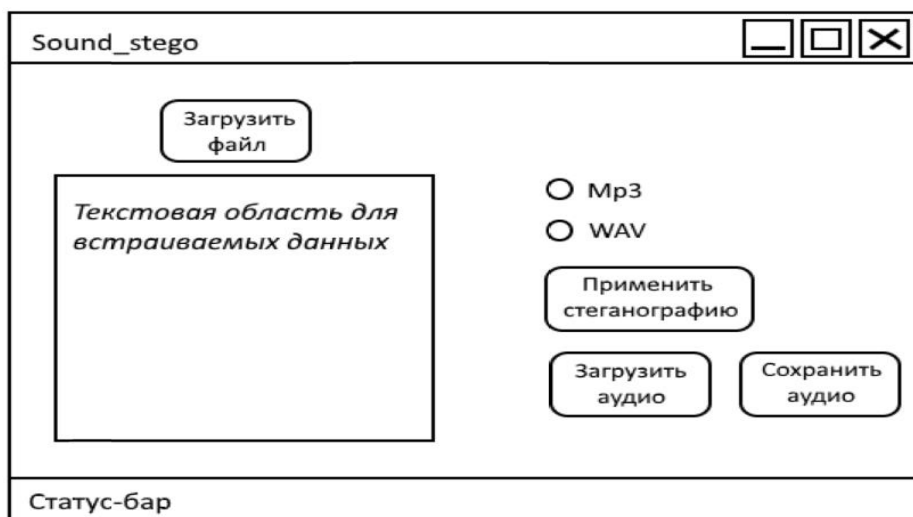


Рисунок 2 – Макет интерфейса программы для аудио стеганографии

Требования к безопасности

Данная программная система может работать с конфиденциальными данными, поэтому программа не должна отправлять эти данные через сеть Интернет или каким-либо образом передавать третьим лицам, а также, по возможности, заниматься шифрованием данных.

Программная система не должна каким-либо образом мешать работе персонального компьютера, изменять файлы, не относящиеся напрямую к программе, а также портить пользовательские файлы.

Программа предназначена для сокрытия информации, но не предусматривает специфических средств защиты в самой программе (не предусмотрена аутентификация, авторизация, регистрация и т.д.).

Проектирование архитектуры системы

Для разработки программной системы «Sound_stego» было решено использовать язык программирования C# и средства разработки VisualStudio.

C# позволяет реализовать интерфейс, а именно WindowsForms. Также с помощью данного языка программирования можно провести основную работу с сигналом, используя объектно-ориентированный подход, при этом он обеспечивает достаточную производительность.

Для понимания как именно пользователь может взаимодействовать с системой была разработана диаграмма прецедентов. Она обеспечивает прозрачное представление системы, функционал и ее подсистемы. Диаграмма прецедентов изображена на рисунке 3

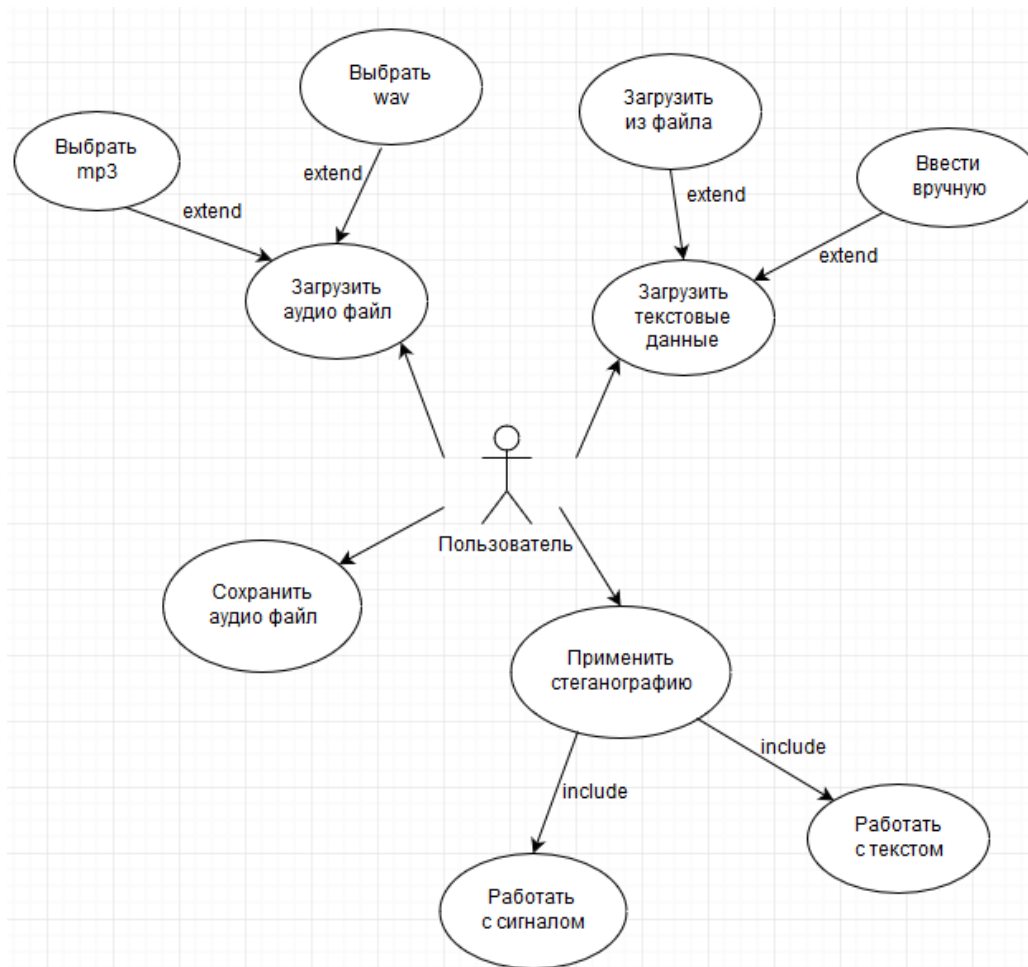


Рисунок 3 – Диаграмма прецедентов приложения для аудио стеганографии

В дальнейшем планируется разработка диаграммы классов приложения и основных алгоритмов работы программы. В качестве контейнеров для сокрытия информации используются форматы .mp3 и .wav. Также должны быть разработаны тесты. Программное обеспечение для аудио стеганографии может быть использовано в качестве учебного стенда для демонстрации принципов работы стеганографических средств защиты информации.

Выводы

В рамках статьи был выполнен обзор и изучение открытого и закрытого программного обеспечения для применения в аудио стеганографии. Спроектирована и начата реализация программного обеспечения для цифровых водяных знаков в аудио файлах.

Система предоставляет возможность применять стеганографию к аудио файлам. Программная система также позволяет загружать исходный аудио файл, загружать встраиваемый текст из файла и сохранять изменённый аудио файл. Поддерживается режим ввода встраиваемого текста сразу в поле ввода.

Литература

1. Стеганография//Wikipedia [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/Стеганография>
2. WAV//Wikipedia.[Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/WAV>
3. MP3 //Wikipedia. [Электронный ресурс]. – Режим доступа:

<https://ru.wikipedia.org/wiki/MP3>

4. Голубев Е.А., Емельянов Г.В. Стеганография как одно из направлений обеспечения информационной безопасности // Технологии информационного общества, Спецвыпуск Т-Comm, 2009. —сС 185-186.

5. Прячем текст в MP3 //Хабрахабр. [Электронный ресурс]. – Режим доступа: <https://habr.com/post/112914/>

6. Грибунин В. Г., Оков И. Н., Туринцев И. В. Цифровая стеганография. - М.: Солон-Пресс, 2002.

УДК 004.89:004.4

ВОЗМОЖНОСТИ ПОИСКА ДОКУМЕНТОВ ОБ ОБРАЗОВАНИИ В ПРОГРАММНОМ КОМПЛЕКСЕ «ГОСУДАРСТВЕННЫЙ РЕЕСТР СВЕДЕНИЙ О ДОКУМЕНТАХ ОБ ОБРАЗОВАНИИ»

Зуев В. М., Рубан О. И., Мартынов О. И., Кравченко Т. М.

Институт проблем искусственного интеллекта

E-mail: gu_ipii@mail.ru

Аннотация:

Рубан О.И., Кравченко Т.М. Возможности поиска документов об образовании в программном комплексе «Государственный реестр сведений о документах об образовании». Рассмотрены основные требования к разработке страницы. Определены технологии, используемые при разработке. Описаны основные компоненты, присутствующие на странице. Приведено краткое описание алгоритма поиска.

Annotation:

Ruban O.I., Kravchenko T.M. Possibilities of searching for documents on education in the program complex "The State Register of Information about Documents on Education". Main requirements for page developing is shown. Technologies used in the development is shown. Main components on the page is described. A brief description of the search algorithm is given.

Введение

Данная задача представляет собой подзадачу проекта «Государственный реестр сведений о документах об образовании» - реестра документов об образовании, служащего для ведения учета об образовательных документах различных организаций. В данной системе имеют возможность зарегистрироваться все образовательные организации нашей республики, причем в зависимости от их уровня меняются функциональные возможности. Поставленная задача по поиску документов позволит ускорить и улучшить доступ к данным о выпускниках республики.

Общая постановка задачи

Необходимо реализовать поиск списка документов об образовании по различным критериям. Поиск должен быть интуитивно понятным и в то же время обеспечивать возможность указывать разные фильтры, как, по известным сведениям, об образовательной организации, так и по сведениям о непосредственно самом выпускнике. Кроме того, необходимо предусмотреть ряд необязательных для заполнения полей. Среди них: Форма обучения, Населенный пункт, Направление подготовки, Полное и сокращенное название организации, Дата рождения и Программа подготовки. Также необходима возможность экспорта данных в Microsoft Excel для дальнейшей печати или других целей.

Реализация

В качестве программной среды разработки был выбран следующий стек технологий: Microsoft SQL Server 2017 – база данных для хранения списка выпускников и их образовательных документов, C# - серверный язык программирования, реализующий связь базы с интерфейсов работы пользователя, HTML5+bootstrap+CSS+JS – графическое отображение клиентской части веб-приложения.

Данная страница позволяет осуществлять поиск документов об образовании по заданным критериям. Для этого предусмотрены три блока с фильтрами поиска: Настройка

фильтрации по данным о документе, о выпускнике, а также опциональные поля, такие как – населенный пункт, направление подготовки, полное и сокращенное название образовательной организации и некоторые другие (рис. 1,2).

Список документов об образовании

Настройки фильтрации по данным о документе Скрыть

Программа подготовки:

Направление подготовки:

Форма обучения:

Год выпуска:

- Все
-
- с по

Рис. 1. Фильтры для поиска документов об образовании. Блок данных о документе

Список документов об образовании

Настройки фильтрации по данным о документе Показать

Настройка фильтрации по данным о выпускнике Скрыть

Фамилия

Год рождения:

- Все
-
- с по

Опциональные поля: Показать

Рис. 2. Фильтры для поиска документов об образовании. Блок данных о выпускнике

После задания определенных критериев, пользователь получает данные в виде таблицы (рис. 3).

Год выпуска:

с по

Настройка фильтрации по данным о выпускнике Показать

Опциональные поля: Показать

[Загрузить список документов](#) [Сброс параметров](#) [Экспорт в Excel](#)

№	Фамилия	Имя	Отчество	Подробнее	Год выпуска	Серия документа	Номер документа
1	Астафьев	Вячеслав	Андреевич	подробно	2010		
2	Шабаленко	Наталья	Геннадиевна	подробно	2010		
3	Бореева	Раиса	Олеговна	подробно	2010		

Рис. 3. Выпускники, получившие документ об образовании в 2010 году

Для просмотра всех данных любого документа из таблицы предусмотрена кнопка «подробно», при нажатии на которую появляется форма с перечнем всех полей базы данных, имеющих отношение к данной записи (рис. 4).

The screenshot shows a web form titled "Информация о выпускнике" (Graduate Information). The form contains the following fields and values:

Field	Value
ФИО	Новицкая Оксана Антоновна
Дата рождения:	01.09.1998
Тип документа:	Аттестат о среднем общем образовании (Аттестат 11 класс)
Тип диплома:	Обычный
Условие зачисления:	Вне конкурса
Квалификация:	---
Серия формы:	ДА
Номер формы:	1800018
Серия документа:	ДА
Номер документа:	1800018
Дата выдачи документа:	12.06.2018
Дата поступления:	01.09.2007
Дата выпуска (окончания обучения, регистрации документа):	31.05.2018
Регистрационный номер документа:	18
Серия приложения:	ДА
Номер приложения:	00018

A "Закрыть" (Close) button is located at the bottom right of the form.

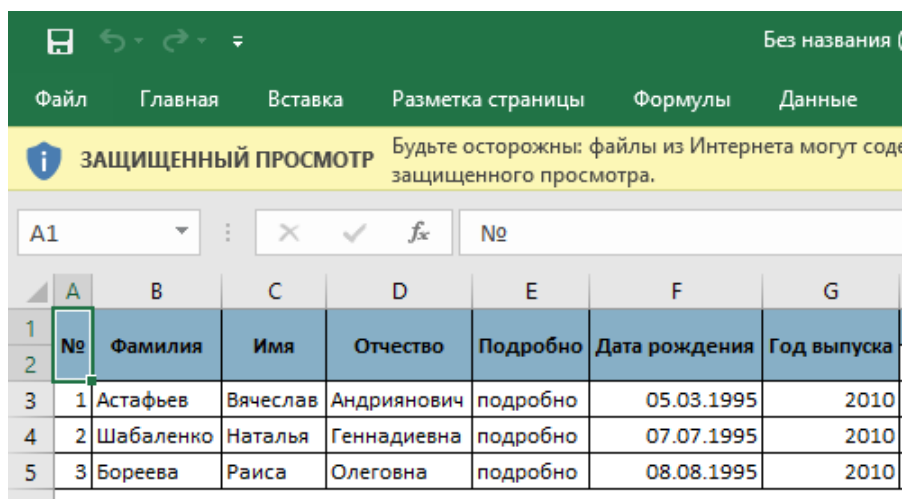
Рисунок 4 - Форма "информация о выпускнике"

Кнопка «Экспорт в Excel» позволяет экспортировать данные, представленные в виде таблицы в новый документ типа .xls (рис. 5).

Кроме поиска по заданным критериям можно получить сведения обо всех выпускниках, не применяя никаких фильтров.

Ниже описана программная реализация алгоритма поиска данных.

После выбора желаемых фильтров, а также дополнительных полей, которые пользователь желает видеть в итоговой таблице, нажимается кнопка «Загрузить список документов». Чтобы пользователь не допускал ошибок при вводе данных, предусмотрена валидация полей.



№	Фамилия	Имя	Отчество	Подробно	Дата рождения	Год выпуска
1	Астафьев	Вячеслав	Андриянович	подробно	05.03.1995	2010
2	Шабаленко	Наталья	Геннадиевна	подробно	07.07.1995	2010
3	Борева	Раиса	Олеговна	подробно	08.08.1995	2010

Рисунок 5 - Фрагмент таблицы списка Выпускников в Excel

После этого вызывается скрипт на языке JavaScript. Этот скрипт собирает все данные о фильтрах, которые выбрал пользователь в одной переменной, которая с помощью Ajax-запроса отправляется в контроллер.

Контроллер, который вызван Ajax-запросом, получает данные из скрипта, а также собирает данные об организации пользователя, и формирует итоговую выборку данных из всех таблиц, которые необходимо вывести. Затем из выборки удаляются те значения, которые не подходят фильтрам, выбранными пользователем.

После успешного выполнения алгоритма контроллера, продолжается работа скрипта, который получает выборку данных из контроллера, и вызывает функцию, которая заполняет таблицу на странице, выводя столбцы по умолчанию, а также те, которые выберет пользователь в опциональных полях.

Выводы

Результатом выполнения поставленной задачи является веб-страница, с помощью которой пользователь может осуществлять поиск с указанием определенных критериев документов об образовании выпускников учебных заведений различной квалификации.

Для удобного доступа к полученной информации также реализован экспорт табличных данных в Microsoft Excel.

Для полноценной работы данной веб-страницы требуется пополнение базы данных новыми сведениями о выпускниках, которые будут использованы в алгоритмах поиска и вывода.

Литература

1. Эспозито, Д. Программирование с использованием Microsoft ASP.NET 4 [Текст] /, 2012.
2. Современный учебник JavaScript [Electronic resource] / Интернет-ресурс. - Режим доступа : [www/ URL: https://learn.javascript.ru/](http://www/URL:https://learn.javascript.ru/) - Загл. с экрана.
3. Документация по Bootstrap 4 [Electronic resource] / Интернет-ресурс. - Режим доступа : [www/ URL: https://bootstrap-4.ru/](http://www/URL:https://bootstrap-4.ru/) - Загл. с экрана.

УДК 004.422.81

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ УЧЁТА И ЖУРНАЛИРОВАНИЯ ДЛЯ СПОРТИВНЫХ КОМПЛЕКСОВ

Полуденный Н. И.

Донецкий национальный технический университет, г. Донецк
кафедра программной инженерии
E-mail: nik.poludenny@mail.ru

Аннотация:

Полуденный Н.И. Программное обеспечение учёта и журналирования для спортивных комплексов. Рассмотрены существующие системы учёта для спортивных комплексов. Определены требуемые данные для представления необходимой и наиболее полной информации в данной системе. Определён подход в формировании требований к системам учёта и оборудованию для системных модулей программного обеспечения.

Annotation:

Poludennyi N.I. Accounting and logging software for sports complexes. Existing accounting systems for sports complexes are reviewed. The required data for the presentation of the necessary and most complete information in this system. The approach to the formation of requirements for accounting systems and equipment for the system modules of software has been determined.

Общая постановка проблемы

Ведение учёта в любой сфере предоставления услуг – вещь необходимая, так как позволяет проследить динамику изменения спроса на предоставляемые услуги и в соответствии с ними скорректировать предложение для извлечения максимальной эффективности. Автоматизация этого процесса с сопровождением хорошо спроектированной базы данных повышает эффективность ведения учёта, в том числе благодаря исключению непреднамеренных и преднамеренных внесений неправильных данных людьми, уменьшению объема записей благодаря структуре реляционной базы данных и т.д. Также можно выделить такие преимущества автоматизации учёта как: перенесение части работы на машины, что позволяет работать удаленно, и это позволяет получать в любой момент времени оперативные данные с сервера.

Исследование существующего программного обеспечения

Объектом исследования являются существующие системы ведения учёта, исследование их структур и модулей, а также создание собственной системы учёта с применением клиент-серверной архитектуры. В настоящее время всё больше аспектов в любой деятельности переходит под автоматическое управление с помощью машин, и ведение учёта один из этих аспектов.

Ведение цифрового учёта имеет массу преимуществ, например:

- при централизации носителя учётной информации в любой момент можно получить наиболее оперативные данные;
- благодаря самой структуре реляционной базы данных исключаются ошибки неверного сопоставления данных в записях;
- ускоряется поиск нужных данных при большом количестве записей;
- повышается безопасность хранения данных и разграничение прав доступа;
- уменьшается вероятность ошибки при ведении учёта;

- возможность работать с записями удалённо;
- снижение бумажного оборота документов (если таковые имеются).

Программная система разделена на две части: серверную и клиентскую. Серверная часть установлена на сервере, который принадлежит спортивному комплексу, а клиентская часть устанавливается на персональные компьютеры пользователей.

На данный момент существует довольно много систем учёта для работы спортивных комплексов. Ниже приведён список подобных по функциональности приложений:

- FitnessPro;
- Тренерская online;
- Fitness365

Программу «FitnessPro» [1] можно использовать как в одном клубе, так и в сети клубов. С помощью синхронизации данных можно увидеть всех клиентов, внесённых в одном клубе – во всех заведениях сети, как и их клубные карты и абонементы. Клиент сможет посещать всю сеть клубов, при этом информация о его абонементе будет видна во всех клубах: название карты, на что распространяется, количество оставшихся посещений по карте и т.д.

«Тренерская online» [2] это интернет-сервис, который решает проблемы спортивных тренеров при подготовке спортсменов. Позволяет вести учёт посещаемости и оплаты занятий, планировать тренировки, составлять анкеты, следить за прогрессом в физическом развитии и вести историю заметок о спортсменах и многое другое. Тренерская упрощает и упорядочивает работу по организации тренировочного процесса, по подготовке и ведению сопутствующей документации.

Fitness365 [3] – настольная программа фитнес клуба. Учет клиентов и их посещений. Их учёт производится с помощью электронных карт. В системе поддерживаются:

- продажа карт;
- продажа абонементов;
- расписание и запись;
- рассылки;
- скидки;
- аналитика;

Клиенты спортивного клуба в любой момент времени могут узнать какие группы доступны для посещения, какие тренеры ведут эти группы и узнать контактную информацию для связи с тренерами. Владение личным счётом позволяет клиентам сразу оплатить запланированное количество посещений занятий и следить за текущим балансом счёта.

Разрабатываемый программный продукт должен соответствовать таким требованиям:

- клиентская часть ПО должна легко устанавливаться на ПК или вообще быть переносной и не требовать установки;
- клиентская часть должна иметь доступ в интернет для связи с сервером;
- должно быть минимизировано количество информации, передаваемое от сервера к клиенту;
- система должна обрабатывать исключительные ситуации;
- пользователя должна достигать только та информация, которая доступна ему для чтения и не нарушает права доступа;
- клиенты спорткомплекса и тренеры должны проходить аутентификацию;
- серверная часть должна брать на себя большую часть вычислительных операций.

Требования к вычислительной мощности сервера можно оценить как средние или выше средних (по меркам серверов), так как один сервер рассчитан на работу одного

спортивного комплекса, а это примерно 20-40 тренеров и до 700-1000 клиентов спортивного комплекса.

Разрабатываемое программное обеспечение рассчитано на широкий круг потребителей, пользовательский интерфейс должен быть максимально упрощен. Допускается использовать пиктограммы для обозначения элементов управления, но в таком случае должны быть организованы всплывающие подсказки о назначении конкретного элемента управления.

База данных, к которой есть возможность доступа у большого числа людей, должна иметь разграничение прав доступа для каждой из ролей. В базе данных разрабатываемого ПО три роли: клиент спорткомплекса, тренер, администратор. Были выделены следующие требования к безопасности:

- на сервере должны быть разграничения прав доступа для чёткого определения возможностей каждой из ролей и тем самым не оставлять возможности несанкционированного доступа к данным злоумышленниками при попытке применить SQL инъекции;
- каждый администратор, тренер и клиент должны иметь собственный аккаунт для идентификации в системе;
- пользователи должны получать информацию из виртуальных таблиц (представлений) базы данных;
- изменение данных в таблицах по возможности должно происходить через функции, которые определены в базе данных;
- ПО должно быть устойчивым к сбоям;
- должны быть организованы атомарные транзакции для обмена информацией сервера и клиента.

Проектирование системы

База данных состоит из двенадцати таблиц определённым образом связанных между собой (рис. 1)

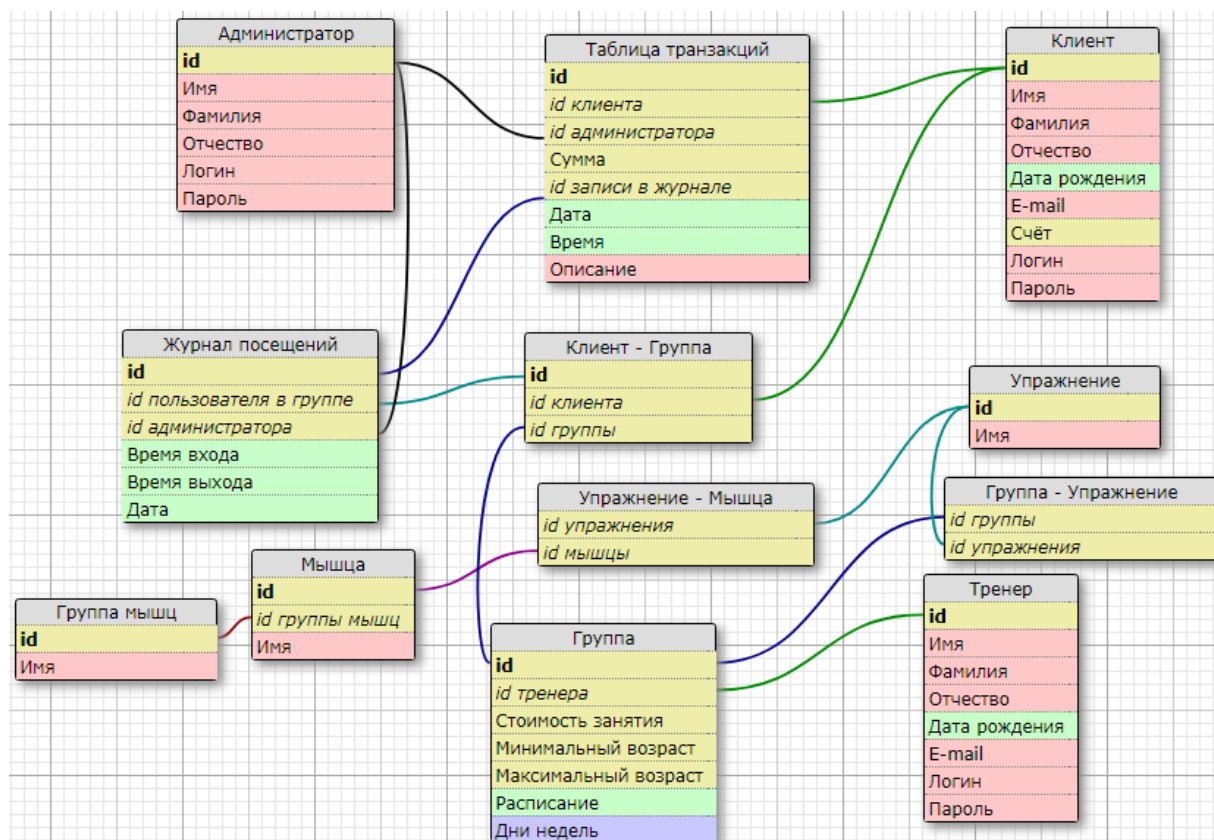


Рисунок 1 - Схема данных базы данных программной системы

Таблицы «Администратор», «Тренер» и «Клиент» служат для хранения данных о каждом пользователе с соответствующей ролью. Поле «Счёт» в таблице клиентов отображает текущее количество денег на счету пользователя, с которого списываются деньги на оплату занятий и на возможные штрафы. Все операции манипуляций с денежными счетами записываются в таблицу «Таблица транзакций» - там сохраняется ссылка на клиента, на администратора проводящего данную транзакцию, дата, время, сумма, если транзакция является оплатой занятия, то также сохраняется ссылка на запись в журнале посещений, если же это штраф или добавление денег на счёт – ссылки на запись в журнале посещений нет, и дополнительные поля для замечания к транзакции. Таблица «Журнал посещений» сохраняет информацию о посещении клиентами заведения, а именно: ссылка на идентификатор записи пользователя в группе, ссылка на администратора, который зарегистрировал посещение спортивного клуба клиентом, время прихода на занятие и время ухода с него и дату.

Таблица «Группа» содержит информацию о группах, созданных тренерами для ведения занятий. Запись хранит информацию о идентификаторе группы, ссылки на тренера, стоимость разового занятия для клиента, возрастные ограничения, а также поля «Расписание» для хранения массива часов и поле «Дни недели» для хранения дней недели в виде массива переменных типа boolean на основе которых отображается расписание, понятное для рядового пользователя. Создавая группу, тренер выбирает какие упражнения будут проводиться, для этого существует таблица «Группа – Упражнение», которая обеспечивает связь «много-ко-многим» и, соответственно, позволяет связать множество упражнений с группой и множество групп с одним упражнением. Таблица «Упражнение – Мышца» связывает упражнение мышцами, которые упражнение нагружает что позволит тренеру составлять оптимальные группы упражнений для своих клиентов.

Таблица «Клиент – Группа» связывает клиентов с тренерской группой, когда клиент оформляет подписку на группу, таким образом ведётся подсчёт количества подписавшихся

на группу и высчитывается потенциальная прибыль с каждой группы, что в свою очередь позволяет делать выводы о спросе и создавать соответствующие предложения.

«Клиентская» версия клиентского приложения может быть установлена на любой персональный компьютер. Пользователи с него могут зарегистрироваться для создания собственного аккаунта (рис. 2, 3). «Администраторская» версия устанавливается только на машины администраторов в спортивном комплексе. Каждый администратор может оттуда зарегистрировать собственный аккаунт. Также только администраторы могут зарегистрировать новых тренеров и создать для них аккаунты, а сама «Тренерская» версия может быть установлена на любой ПК, но для входа в систему потребуется пара логин-пароль одного из аккаунтов с ролью тренера.

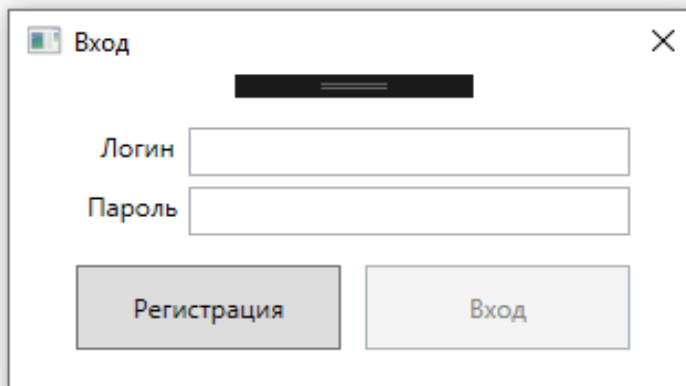


Рисунок 2 - Окно аутентификации для «пользовательской» и «администраторской» версий приложения

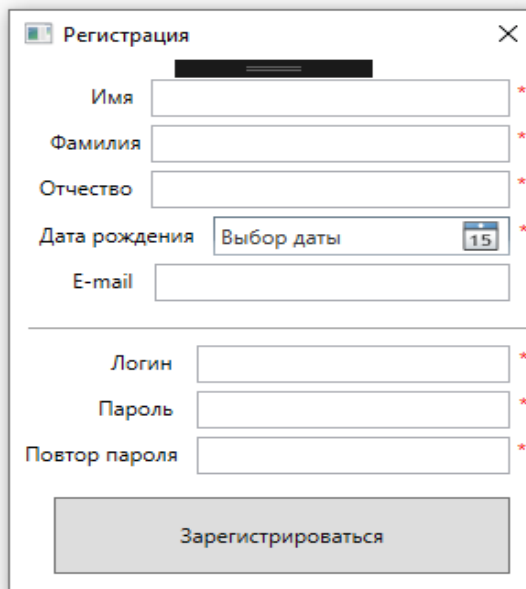


Рисунок 3 - Окно регистрации для пользователя

Правильность введённых данных обрабатывается на стороне сервера: при отсутствии записи с введённым логином или несоответствием логину пароля вернутся соответствующие коды ошибок.

Выводы

Все системы учёта направлены на упрощение предоставления информации и выполнения различных операций над ней и сейчас всё больше и больше компаний в различных сферах бизнеса прибегают к автоматизации учёта. Внедрение автоматизированного учёта позволяет вести мониторинг всей системы и отслеживать эффективность работы тех или иных элементов системы и предпринимать соответствующие действия для повышения дохода [4,5]. Данная система создана для ведения учёта и журналирования в спортивные комплексы и позволяют клиентам комплекса планировать свои занятия, содержать свой счёт для оплаты занятий и предоставляет контактную информацию тренеров. Тренерам спортивных комплексов система позволяет планировать расписания своих групп, указывать возрастные ограничения, задавать цену занятия и указывать упражнения, которые будут выполняться группой. Ведение журнала посещений клиентов позволяет отслеживать их активность и предоставляет им отслеживать историю оплат своих занятий.

Литература

1. Программа для автоматизации фитнес клуба [Электронный ресурс] / Интернет-ресурс. – URL: <https://as-service.com.ua/light-soft/avtomatizatsiya-fitness-klubov.html>
2. Спортивный журнал тренера «Тренерская online» [Электронный ресурс] / Интернет-ресурс – URL: <https://tronline.ru>
3. Fitness365 [Электронный ресурс] / Интернет-ресурс – URL: <https://www.fitness365.ru>
4. Карпов А. А. Эффективная система бюджетирования и управленческого учета. Практические советы и рекомендации [Текст] / Карпов А. А.
5. Универсальная система учёта [Электронный ресурс] / Интернет-ресурс. – URL: <http://ususoft.com.ua/index.php>

СЕКЦИЯ 3

СИСТЕМЫ
АВТОМАТИЗИРОВАННОГО
ПРОЕКТИРОВАНИЯ

ИУС
МКМ 2019

УДК 004.021

ИССЛЕДОВАНИЕ ТЕХНОЛОГИЙ СОЗДАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ НАДСТРОЕК НАД ВИЗУАЛЬНЫМИ РЕДАКТОРАМИ ТИПА VISIO

Ржевский К.В., Григорьев А.В.

Донецкий национальный технический университет
кафедра программной инженерии

E-mail: nory4ik4@mail.ru, grigorievalvl@gmail.com

Аннотация:

Ржевский К.В., Григорьев А.В. Исследование технологий создания интеллектуальных надстроек над визуальными редакторами типа Visio. Статья посвящена вопросу создания интеллектуальных надстроек над визуальными редакторами типа Visio.

Annotation:

Rzhevsky K.V., Grigoriev A.V. The study of technologies to create intelligent add-on visual editors such as Visio. The article is devoted to the creation of intelligent add-on visual editors such as Visio.

Ключевые слова: Microsoft Visio, интеллектуальные надстройки, САПР.

Актуальность работы: заключается в необходимости исследования создания технологии интеллектуальных надстроек над визуальными редакторами.

Цель работы: исследование и анализ функциональных возможностей и путей расширения функционала программного обеспечения для работы с блок-схемами.

Научная новизна: выполнен анализ существующих систем для построения блок-схем; на основе визуальных компонентов и библиотек Visio, а также платформы Microsoft Visio, исследованы технологии создания интеллектуальных надстроек над визуальными редакторами.

Практическое значение полученных результатов состоит в следующем: Исследованы технологии создания надстроек над визуальными редакторами блок-схем.

Введение

Эффективность процессов проектирования в любых предметных областях в настоящее время во многом определяется наличием и степенью развития средств, позволяющих проектировщику формализовать и повторно использовать экспертные методики для проектирования тех или иных изделий. Методики проектирования призваны существенно упростить проектировщику решение задачи проектирования, исходя из требований технического задания на изделие (ТЗ). Такие средства позволяют создавать методики проектирования, сохранять их, редактировать и - применять по мере надобности. Эффект в этом случае состоит в сокращении времени и трудоемкости процесса принятия проектных решений. Учитывая актуальность задачи построения инструментальных средств создания моделей методик проектирования, важной представляется задача анализа существующих подходов в этой области, а так же определение перспективных направлений их дальнейшего развития.

1. Специфика задания методик проектирования как алгоритмов проектирования в современных САПР

Краткая характеристика:

- 1) наличие спец. языка программирования, как средства описания порядка синтеза графической модели (комплекса чертежей) объекта проектирования некоторого типа;
- 2) алгоритм описывает порядок действия проектировщика при построении (проектировании) по ТЗ блока заданного типа как 3(2)d-модели;
- 3) алгоритм может создаваться, сохраняться, редактироваться, вызываться на выполнение;
- 4) алгоритм – это модуль знаний;
- 5) набор внешних переменных алгоритма рассматривается как ТЗ модуля знаний; ТЗ можно создавать явно, сохранять, редактировать, использовать повторно;
- 6) рассматривается набор внутренних, инкапсулированных расчетных данных;
- 7) создается набор формул (однонаправленных) для расчета инкапсулированных переменных по исходным данным;
- 8) для синтеза графической модели используется набор команд типа AutoCAD для создания трех(двух)-мерной модели объекта заданного типа по заданному ТЗ по расчетным переменным;
- 9) язык включает набор команд IF, WHILE, FOR для управления процессом рисования;
- 10) язык имеет команды «добавить» и «удалить» для редактирования компонент трехмерной модели.

Практическое применение: алгоритмы проектирования как языковые средства для создания, хранения и вызова методик проектирования, фактически является современным общепринятым инструментом в наиболее развитых САПР (САТИА, ГРАЦИЯ и т.д.).

Достоинства:

- 1) использует полноценный язык программирования, отвечающий задачам автоматизации проектирования;
- 2) обеспечивает возможность прописать все этапы методик проектирования как набор последовательных действий проектировщика;
- 3) рассматривает явно определенное ТЗ;
- 4) позволяет косвенно описать тип блока как множество альтернативных структур объекта проектирования, имеющих структурные отличия во внешних структурных связях, внутренних компонентах и – их структурных связях.

Недостатки:

- 1) ориентация только на высококвалифицированного эксперта-проектировщика, способного явно сформировать алгоритм проектирования;
- 2) фиксированный набор данных, составляющих любое ТЗ;
- 3) модель структуры объекта не достаточно явно прописана как набор блоков и – структурных связей, а рассматривается как набор геометрических примитивов;
- 4) нет вложенных алгоритмов проектирования;
- 5) отсутствует возможность явно описать тип блока как множество альтернативных структур объекта проектирования, имеющих структурные отличия во внешних структурных связях, внутренних компонентах и – их структурных связях.

В качестве промежуточного вывода можно отметить, что подобный подход может применяться в рамках любого САПР.

2. Базовый функционал визуального редактора Visio

Microsoft Visio — редактор диаграмм и блок-схем для Windows.

Выпускается в трёх редакциях: Standard, Professional и Pro for Office 365.

Аналогично с Adobe Reader в стандартный набор программ MS Office входит только средство для просмотра и печати диаграмм Microsoft Visio Viewer. Полнофункциональная версия Microsoft Visio Professional для создания и редактирования монограмм и диаграмм в пакеты MS Office не входит и распространяется отдельно.

Первоначально Visio разрабатывался и выпускался компанией Visio Corporation.

Microsoft приобрела компанию в 2000 году.

Когда продукт назывался Visio 2000, был выполнен ребрендинг, и продукт был включён в состав Microsoft Office. Последняя версия продукта была выпущена в 2016-м году [1].

Редактор предназначен для создания диаграмм и блок - схем.

В качестве его функциональных возможностей можно назвать:

- создание диаграмм;
- создание шаблонов;
- создание сценариев использования продукта;
- создание блок – схем.

3. Анализ существующих средств автоматизации работы в рамках визуальных редакторов типа Visio

Функциональные возможности Microsoft Visio можно расширить следующими способами:

- Создание специальных макросов и надстроек Visio.
- Создание надстроек COM (компонентной объектной модели).
- Использование элемента управления документом Visio.
- Использование библиотеки типов Visio.

Рассмотрим эти возможности детальнее.

3.1. Макросы и надстройки

Макросы и надстройки — это программы, расширяющие функциональные возможности Visio. Именно то, как вы запускаете макрос или надстройку, зависит от контекста, для которого она была разработана.

Если надстройка предназначена для запуска приложения Visio, ее следует выполнять как любую программу Microsoft Windows (например, дважды щелкнув значок на рабочем столе).

3.2. Надстройки COM

Начиная с Visio 2002, вы можете использовать надстройки COM одинаково стандартизированным способом, как и в других приложениях Microsoft Office. Надстройка COM должна быть зарегистрирована в приложении Visio и может работать в нескольких приложениях. Например, вы можете создать надстройку COM, которая выполняет одну и ту же задачу в Visio и Microsoft Excel, а также в любом приложении Microsoft Office. Вы можете создавать надстройки COM с помощью Microsoft Visual Basic 5.0 и более поздних версий, Microsoft C++, Microsoft Office 2000 Developer Edition и более поздних версий, а также любых приложений Microsoft Visual Studio .NET [2].

3.3. Элемент управления документом Visio

С помощью элемента управления документом Visio вы можете интегрировать поверхность рисования Visio в приложение.

Элемент управления документом Visio — это элемент управления ActiveX, который предоставляет полный доступ к объектной модели Visio (API) и пользовательскому интерфейсу для интеграции пользовательского интерфейса Visio, настройки внешнего вида и

автоматизации Visio в приложениях. Элемент управления документом Visio предоставляется вместе с Microsoft Visio 2013 и устанавливается при установке Visio [3].

3.4. Сведения о библиотеке типов Visio

Продукты Visio включают библиотеку типов, определяющую объекты, свойства, методы, события и константы, которые Visio предоставляет клиентам автоматизации. Чтобы использовать библиотеку типов Visio, среда разработки должна ссылаться на нее. Проект Visual Basic для приложений (VBA) документа Visio автоматически ссылается на библиотеку типов Visio. В других средах разработки для ссылки на библиотеку необходимо выполнить соответствующие действия.

Имена библиотек, на которые ссылаются проекты VBA, отображаются в списке проект/Библиотека в обозревателе объектов в редакторе Visual Basic.

Библиотека типов полезна по следующим причинам:

- Сведения в библиотеке типов служат входными данными для браузеров объектов, предоставляемых VBA и другими средами разработки. Вы можете использовать браузеры объектов для просмотра описаний объектов, предоставляемых серверами автоматизации (например, приложением Visio), установленными в системе. Например, можно просмотреть синтаксис свойства, метода или события Visio и вставить код, отображаемый браузером, в программу.
- Библиотека типов позволяет средам разработки привязывать код программы к серверному коду автоматизации при компиляции (разработки), а не динамически во время выполнения. В результате программа часто работает быстрее. Например, можно использовать Visio.Page, Visio.Shape, Visio.Document и т.д. вместо Object [4].

Т.о., в качестве вывода по функциональным возможностям визуальных редакторов типа Microsoft Visio можно сказать, что данные системы не обладают таким инструментом, как интеллектуальная надстройка, что существенно снижает эффективность их работы.

Выводы.

Исследованы средства расширения функционала пакета Microsoft Visio, выявлены особенности реализации и дальнейшие пути исследования улучшения работы в данной теме. Microsoft Visio является расширяемым инструментом и имеет много возможностей по встраиванию в другое ПО. Был сделан вывод, что в стандартной версии пакета Microsoft Visio нет интеллектуальных надстроек. Однако средства для разработки интеллектуальных надстроек присутствуют и в том числе есть существующие решения, представленные на рынке.

Литература

1. Microsoft Visio [Электронный ресурс] // Википедия – Режим доступа: https://ru.wikipedia.org/wiki/Microsoft_Visio – Загл. с экрана.
2. О расширении функциональных возможностей Visio [Электронный ресурс] // Хабр – Режим доступа: <https://docs.microsoft.com/ru-ru/office/vba/visio/concepts/about-extending-the-functionality-of-visio> – Загл. с экрана.
3. Сведения об элементе управления документом Visio [Электронный ресурс] // Docs Microsoft – Режим доступа: <https://docs.microsoft.com/ru-ru/office/vba/visio/concepts/about-the-visio-drawing-control> – Загл. с экрана.
4. Сведения о библиотеке типов Visio [Электронный ресурс] // Docs Microsoft – Режим доступа: <https://docs.microsoft.com/ru-ru/office/vba/visio/concepts/about-the-visio-type-library> – Загл. с экрана.

УДК 004.94

ПРИМЕНЕНИЕ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ДЛЯ УСОВЕРШЕНСТВОВАНИЯ УПРАВЛЕНИЯ ТРАНСПОРТНЫМИ ПОТОКАМИ ГОРОДА И БЕЗОПАСНОСТИ НА ПЕШЕХОДНЫХ ПЕРЕХОДАХ

Розанова С.А., Косенко Ю.А., Незамова Л.В.
Донецкий национальный технический университет
Кафедра «Программной инженерии»
E-mail: khitrik.rozanova.svetlana@gmail.com

Аннотация:

Розанова С.А., Косенко Ю.А., Незамова Л.В. Применение имитационного моделирования для усовершенствования управления транспортными потоками города и безопасности на пешеходных переходах. В работе построена имитационная модель сложной улично-дорожных развязки города Донецка в среде Anylogic. Продемонстрировано решение проблемы образования пробок на указанном участке дороги и показаны возможности решений задач по оптимизации движения транспорта с помощью имитационного моделирования.

Annotation:

Rozanova S.A., Kosenko Yu.A., Nezamova L.V. The use of simulation modeling to improve the management of city traffic and safety at pedestrian crossings. An imitacional model of a complex street-road junction in the city of Donetsk in the Anylogic environment has been built. Solution of the problem of traffic jams on the specified section of the road was demonstrated and showed the possibilities of solving problems of optimizing traffic through simulation.

Ключевые слова: имитационное моделирование, библиотека дорожного движения, транспортная развязка, проблема пробок, Anylogic.

Введение

На сегодняшний день проблема пробок на дорогах является актуальной как никогда. Рост транспорта на дорогах с каждым годом увеличивается, тогда как дорожная развязка не везде развивается и улучшается. В результате появляются пробки, которые, как известно, способствуют ДТП, ухудшению качества дороги, затруднению движения пешеходов и т.д. В данной статье рассматривается проблема дорожных заторов на Макеевском шоссе, а именно, на кольце в районе «Мотеля».

Целью работы является создание полноценной имитационной модели транспортной системы и оптимизация транспортного потока на указанном участке с помощью программного обеспечения Anylogic. Для достижения поставленной цели были выявлены следующие задачи:

- изучить существующие среды моделирования имитационных моделей;
- провести исследование для формулирования поведения агентов в транспортной системе;
- провести анализ статистических данных;
- спроектировать и реализовать модель транспортной системы;
- сделать вывод по полученным результатам и предложить способы решения данной проблемы.

Имитационное моделирование транспортных потоков с помощью Anylogic

Выбор платформ, подходящих для моделирования различных систем, в том числе транспортных, велик, и он постоянно увеличивается. Для решения поставленной задачи была выбрана среда моделирования Anylogic, поскольку инструментальная система этой платформы не ограничивает разработчика одной парадигмой моделирования, что является характерным для многих существующих на рынке инструментов моделирования. Пользуясь Anylogic, разработчик имеет возможность использовать различные уровни абстрагирования и различные стили, концепции, а также смешивать их при создании одной и той же модели. Однако главной причиной, по которой была выбрана среда Anylogic, является разнообразие библиотеки дорожного движения и action charts (блок-схем), которые активно использовались в данной работе.

Анализ транспортной развязки

В рамках проекта была разработана имитационная модель, позволяющая получить всеобъемлющую статистику системы и воспроизводящая наиболее близкое к реальному поведению агентов.

Но перед тем, как приступить непосредственно к построению имитационной модели – назовем ее «Мотель» - необходимо было собрать реальные данные об автомобилепотоке на рассматриваемом участке дороге в час пик. Был выбран определенный промежуток времени: с 8:00 до 9:00 утра. В течении данного времени фиксировалось количество автомобилей на основных дорогах, таких как: Макеевское шоссе, проспект Ильича, улица Элеваторная и Токмакская. При этом получаемая статистика разделялась на количество частных автомобилей, количество автобусов и маршрутных такси, количество специальной техники и грузовых автомобилей и т.д. (таблица 1).

Таблица 1

Наименование дороги	Частный транспорт	Автобусы и маршрутное такси	Специальная техника и пр.
Макеевское шоссе	1234	432	107
проспект Ильича	1047	69	43
улица Элеваторная	402	61	5
улица Токмакская	398	63	3

Исходя из полученных данных, можно сделать вывод, что в рассматриваемый промежуток времени автомобилепоток со стороны Макеевки больше, чем со стороны Донецка. Это обусловлено тем, что основная часть населения Макеевки едет на работу, учебу и т.д. В результате на Макеевском шоссе образуется пробка (рис. 1).



Рисунок 1 – Появление пробки на модели

При построении имитационной модели особое внимание было уделено логике движения агентов в системе. Опираясь на ПДД была разработана подробная блок-схема, которая отвечает за правильное перемещение частного транспорта и автобусов на кольце. Отдельно была составлена блок-схема движения основного потока машин – частных (рис. 2) – и автобусов (рис. 3). В итоге разработанная имитационная модель является максимально приближенной к реальной.

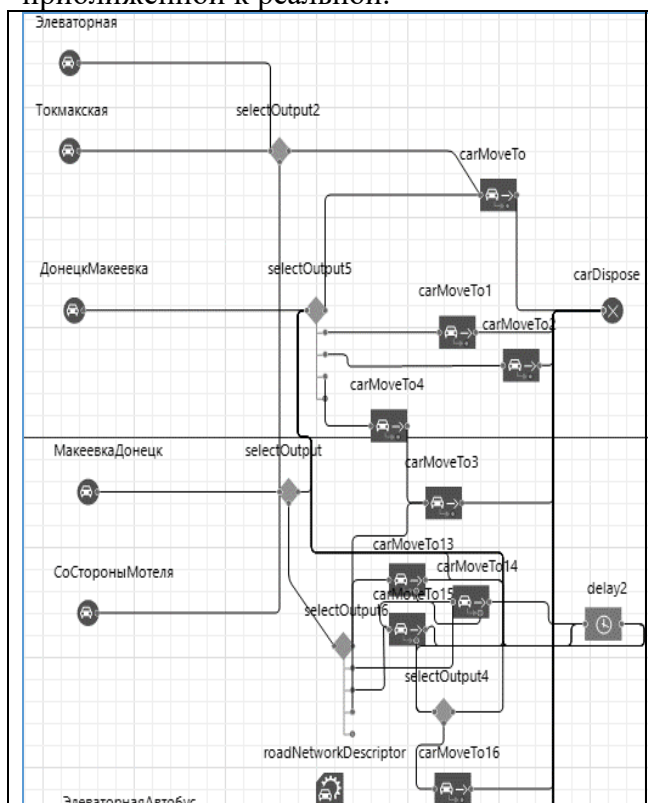


Рисунок 2 – Блок-схема движения частного транспорта

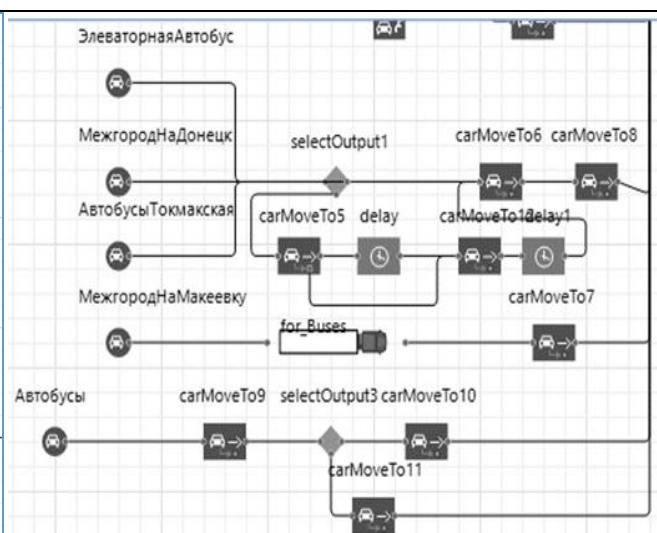


Рисунок 3 – Блок-схема движения автобусов

Для того, чтобы имитировать остановку автобусов, был создан отдельный агент For_buses, в котором была прописана логика движения автобусов (рис. 4).

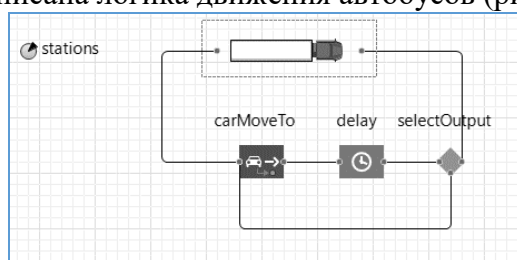


Рисунок 4 – Блок-схема агента For_buses

В результате анализа появилась необходимость в четком регулировании траффика на участке дороги. После проведения нескольких имитационных экспериментов была выдвинута идея установить систему согласованных светофоров, которые бы контролировали перекрестки на кольце. Работа данной системы зависела бы от времени дня: поскольку в час пик утром количество машин на Дорогу Makeyevka-Donetsk больше, чем на Makeyevka-Donetsk, то зеленый свет на светофоре trafficLight1 должен гореть дольше, тогда как на светофоре trafficLight – меньше.



Рисунок 5 – Применение системы светофоров

При создании системы светофоров был проведен эксперимент, в котором производился подбор оптимальных параметров работы каждой фазы того или иного светофора (по умолчанию – по 30 сек). Параметры изменялись в диапазоне от 10 до 35 сек. с шагом 5 сек. В итоге, например, для светофора trafficLight1 время работы зеленого света составляет 30 сек., а красного – 20сек. Для всех остальных светофоров в разработанной системе алгоритм нахождения параметров работы фаз аналогичен. После введения светофорной системы, затор уменьшился, однако проблема пробок осталась. Поэтому было принято решение добавить по одной полосе на дороге проспект Ильича и Макеевское шоссе, а также - перенести автобусную остановку (рис. 6), поскольку она находится довольно близко к перекрестку, вследствие чего водителям автомобилей приходится лишний раз снижать скорость, чтобы пропустить автобус, направляющийся к остановке.



Рисунок 6 – Перенос автомобильной остановки

После введенных изменений дорожная обстановка на рассматриваемом участке транспортной сети значительно улучшилась, что и показано на графике зависимости

количества автомобилей на участке дороги (в данном случае Макеевское шоссе) от времени – ось абсцисс (рис. 7).

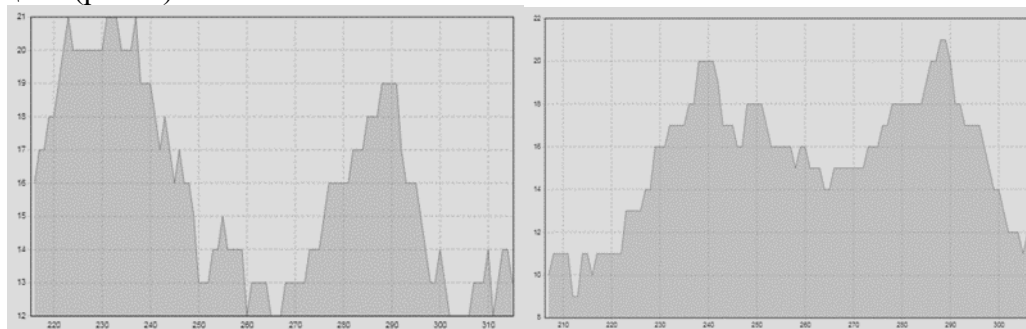


Рисунок 7 – Статистика загруженности дорог

Как видно на рис. 8, после того как количество автомобилей в системе было увеличено на 70%, на Макеевском шоссе снова образуется пробка. Однако разработанная имитационная модель гибко настраиваемая, что позволяет легко ее оптимизировать при любом размере автомобилепотока.



Рисунок 8 – Поведение модели при увеличенном автомобилепотоке

Заключение

В результате проделанной работы было проведено исследование дорожно-транспортной развязки «Мотель» и анализ реализуемых методов оптимизации транспортных потоков, с помощью пакета Anylogic.

Литература

1. Анализ подходов и программного обеспечения для имитационного моделирования социальных и экономических систем: Статья/ А.М. Коровин — Вестник ЮУрГУ, 2012. — 98–100с.
2. Создание простой модели пешеходного перекрестка в программе Anylogic 7.3: Электронный научно-практический журнал / Мочалин А.А. «МОЛОДЕЖНЫЙ НАУЧНЫЙ ВЕСТНИК», 2017.
3. AnyLogic. Многоподходное имитационное моделирование. [Электронный ресурс]. – Режим доступа URL: <http://www.anylogic.ru/>

УДК 004.4'23

3D-ПРИНТЕР, АНАЛИЗ СОВРЕМЕННОГО СОСТОЯНИЯ, ПЕРСПЕКТИВ РАЗВИТИЯ И ПРАКТИЧЕСКОГО ПРИМЕНЕНИЯ

Харламов И.А., Григорьев А.В.

Донецкий национальный технический университет

Кафедра «Программной инженерии»

E-mail: ilya.kharlamov@gmail.com

Аннотация:

Харламов И.А., Григорьев А.В. 3D-принтер, анализ современного состояния, перспектив развития и практического применения. В работе проведен анализ современного состояния, перспектив развития и практического применения 3D-принтеров. Определены перспективы развития программных средств управления 3D-принтерами.

Annotation:

Kharlamov I.A., Grigoriev A.V. 3D-printer, analysis of the current state, prospects for development and practical application. The paper analyzes the current state, prospects for the development and practical application of 3D printers. The prospects for the development of software management 3D-printers.

Ключевые слова: G-code, FDM, 3D-принтер, расходные материалы.

Введение

3D-принтеры сегодня активно входят в моду, что делает актуальной задачей поиска новых областей их применения на практике. Практически все современные 3D-принтеры имеют схожую структуру и общий принцип работы. Следует так же отметить унификацию программных средств поддержки их работы.

Наиболее типичным принтером, доступным любому пользователю сегодня, можно назвать FDM 3D-принтер. FDM расшифровывается как «моделирование методом наплавления» (Fused Deposition Modeling) [1].

Принцип действия FDM-принтера прост: раздаточной головкой на поверхность охлаждаемой платформы-основы выдавливаются капли находящегося в разогретом состоянии термопластика. Быстро застывая и слипаясь между собой, капли формируют слои создаваемого объекта. В итоге получается объемный предмет, имеющий некоторое утилитарное назначение.

Типичность FDM принтера позволяет рассматривать его как предмет анализа.

Целью данной работы является анализ современного состояния, перспектив развития и практического применения 3D-принтеров на примере FDM 3D-принтера.

Структура типичного 3D-принтера

3D-принтер состоит из (см. рис. 2) корпуса (1), закрепленных на нем направляющих (2), по которым перемещается печатающая головка (3) с помощью шаговых двигателей (4), рабочего стола (5), на котором выращивается изделие; и всё это управляется электроникой (6).



Рисунок 1 – Объекты 3D-печати

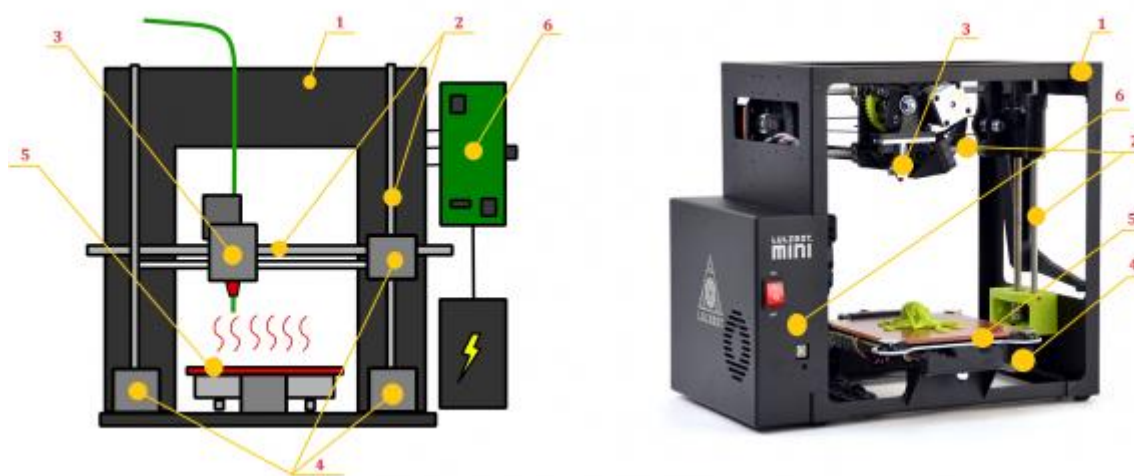


Рисунок 2 – Структура типичного 3D-принтера

Процесс печати 3D-принтера

Расходные материалы (филаменты) для 3D-принтеров представляют из себя пластиковые нити, намотанные на катушки. Расходные материалы бывают различных типов и свойств (см. рис. 3).

Килограмм самого дешевого пластика можно купить за 500 руб., хотя более интересные варианты (например, имитаторы древесины или песчаника с наполнителями из настоящей древесины или камня) уже могут обойтись в несколько раз дороже.



Рисунок 3 – Набор расходных материалов для 3D-печати

Порядок работы 3D принтера

Рассмотрим вкратце принципы работы 3D-принтера (рис. 4). Нить (филамент) (1) поступает в печатающую головку (Экструдер) (2), в которой разогревается до жидкого состояния и выдавливается через сопло экструдера. Шаговые двигатели с помощью зубчатых ремней приводят в движение Экструдер (2), который перемещается по направляющим (3) и наносит пластик на платформу (4) слой за слоем. Снизу в вверх. В итоге ваше изделие (5) растёт слой за слоем [2].

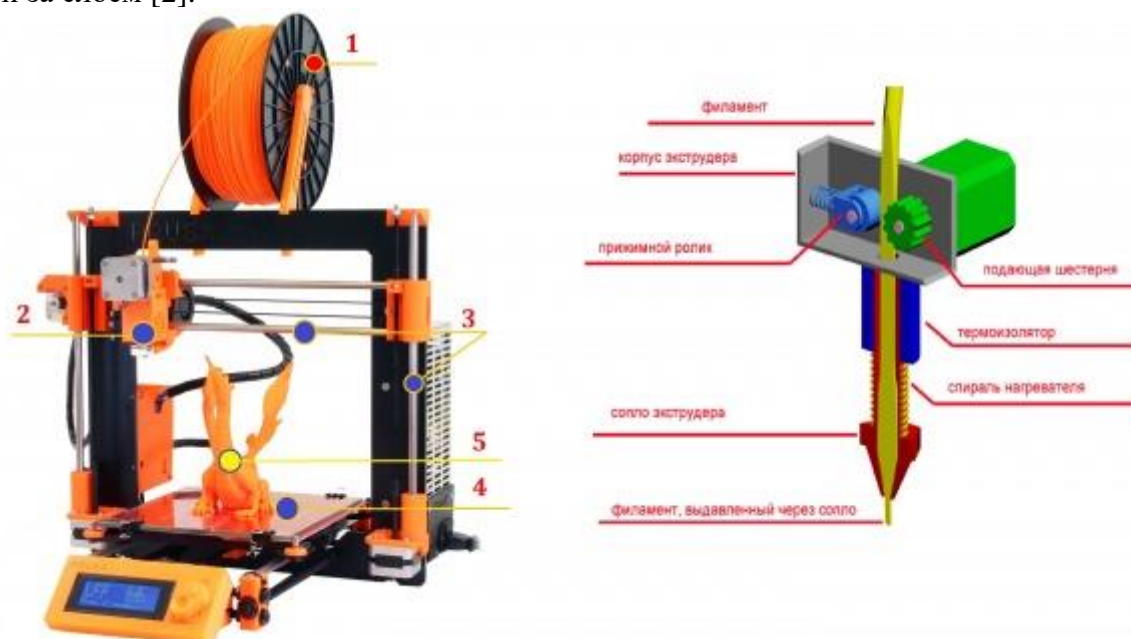


Рисунок 4 – Принцип работы 3D-принтера

Функциональные возможности 3D-принтеров

Все зависит от качества процесса и используемого пластика. На домашнем 3D-принтере, например, можно напечатать рабочие шестеренки для самодельных роботов или пластиковые корпуса для электронных гаджетов. Самое замечательное то, что появится возможность создавать уникальные изделия или ремонтировать вещи, даже те, что сняли с

производства. Себестоимость одной детали, как правило, будет выше, чем у ширпотреба, но и здесь бывают исключения.

Производство настольных 3D-принтеров уже всюду налажено в России, причем отечественные аналоги ничем не хуже западных вариантов. Полного замещения комплектующих пока никому из отечественных производителей добиться не удалось, но готовые продукты дешевле западных конкурентов и не уступают им по характеристикам или качеству печати, а за сервисным обслуживанием не придется далеко бегать. Помимо FDM-принтеров существуют и машины, работающие с жидкими смолами, отверждаемыми светом, пластиковыми и металлическими порошками, спекаемыми лазерами, и даже устройства, изготавливающие высокоточные трехмерные модели из листов обычной бумаги, но это уже отдельная история.

Широта возможностей 3D-принтеров позволяет использовать их в комплексе с современными САПР, применяя их, например, для печати материнских плат компьютеров, программируемых интегральных схем, жилых построек и т.д.

Языки программирования 3D принтеров

Для всех принтеров стандартом программного языка является «G-код». Это стандартный язык для написания программ для устройств с числовым программным управлением (ЧПУ). На данный момент, существует самая последняя версия G-кода – 3.2.0.0. Но уже известны некоторые новые разрабатываемые функции для G-кода версии 3.2.0.7 [3]:

Код модуля	Описание	Статус	на вход подается	на выходе получаем
100	Загрузка изображения	Готово	—	рисунок
101	Загрузка траекторий из DXF файла	Готово	—	векторы
102	Загрузка траекторий из GERBER файла	Готово	—	векторы
103	Загрузка траекторий из PLT (HPGL) файла	Готово	—	векторы
104	Загрузка файла сверловки	Готово	—	векторы
105	Ввод текста	Разработка	—	векторы/рисунок
106	Конструктор простых траекторий	Разработка	—	векторы
107	– больше его нет –	удален	—	векторы
200	Преобразование контуров изображения в векторы	Готово	рисунок	векторы
201	Растровое выжигание	Планируется	рисунок	векторы
202	Преобразование рисунка в точки для выжигания, гравировки	Готово	рисунок	векторы
300	Получение новых контуров со смещением	Готово	векторы	векторы
301	Оптимизация траекторий	Разработка	векторы	векторы

Код модуля	Описание	Статус	на вход подается	на выходе получаем
302	Модификация траекторий (смещение, вращение, изменение пропорций)	Готово	векторы	векторы
303	Векторный редактор	В разработке	векторы	векторы
304	Модификация траекторий для использования флюгерного ножа	Планируется	векторы	векторы
500	Объединение данных из нескольких модулей	Готово	векторы	векторы
900	Генерация G-кода	Готово	векторы	—
901	Управление GRBL контроллером	Планируется	векторы	—

Т.о., G-код постоянно развивается, совершенствуется. Это позволяет все более расширять спектр областей применения 3D-принтеров, расширять функционал и повышать их эффективность.

Заключение

В работе проведен анализ современного состояния, перспектив развития и практического применения 3D-принтеров на примере FDM 3D-принтера. Определены перспективы развития программных средств управления 3D-принтерами.

Литература

1. 3D-принтеры: зачем они нужны и как они работают. <http://zoom.cnews.ru/publication/item/54392>. - Заголовок с экрана.
2. Что такое 3D-принтер. <https://3dtoday.ru/wiki/3Dprinter/>. - Заголовок с экрана.
3. А.С. Пайвин, О.А. Чикова. Основы программирования станков с ЧПУ / Урал. гос. пед. ун-т. – Екатеринбург, 2015. – 102 с.

УДК 004.94

МЕХАНИЗМЫ ДВИЖЕНИЯ И ФИЗИЧЕСКИЕ ЗАВИСИМОСТИ В СОВРЕМЕННЫХ САПР

Чернышов Д.Н., Григорьев А.В.

Донецкий национальный технический университет

Кафедра «Программной инженерии»

E-mail: dima.ch2000@mail.ru

Аннотация:

Чернышов Д.Н., Григорьев А.В. Механизмы движения и физические зависимости в современных САПР. В работе описана возможность протестировать движение деталей на примере САПР Catia v5. Проанализированы некоторые физические расчеты в современных САПР.

Annotation:

Chernyshov D.N., Grigoriev A.V. Movement mechanisms and physical dependencies in modern CAD systems. Possibility of testing moving parts is described using CAD Catia v5. Some physical measurements in modern CAD systems are analyzed.

Ключевые слова: механизмы движения, физические расчеты, прочность, термодинамика, движущиеся детали, Catia v5.

Введение

Многие считают, что среди составляющих системы САПР важнейшим является САД (Computer Aided Design). Что ж, это утверждение справедливо, если проект так и остается в виде чертежей. Однако при переносе в реальные условия такой проект может повести себя непредсказуемо. Поэтому в современных САПР используется комплекс САЕ (Computer Aided Engineering), который обладает той же важностью, что и САД.

Целью данной работы является рассмотрение некоторых возможностей комплекса САЕ, таких как:

- моделирование движения тел,
- ограничения при построении,
- расчеты прочности изделия,
- термодинамические расчеты.

Моделирование движения деталей в механизмах

Ни для кого не секрет, что сегодняшний виртуальный мир немислим без анимации. Напомню, что анимация — это технология, позволяющая при помощи неодушевленных неподвижных объектов создавать иллюзию движения. Для создания мультипликации, рекламы, фильмов и другого анимационного цифрового контента применяются специализированные программные продукты. Системы автоматизированного проектирования также обладают средствами для визуализации движения механизмов, приборов, устройств, узлов машин.

Проект анимации — это «пошаговая стратегия», последовательная комбинация шагов, каждый из которых представляет собой действие одного или нескольких компонентов механизма в пространстве в соответствии с законом движения. [1]

Задание ограничений при построении проекта

Было бы очень неприятно, если бы наш проект при воспроизведении анимации просто развалился на части. Для этого в САПР существуют ограничения. Рассмотрим некоторые из них в среде Catia v5.

В режиме Sketcher существуют размерные и геометрические ограничения. Они служат для математического закрепления геометрии в пространстве координат. Без наложения ограничений созданный примитив становится непредсказуемым и внесение в модель изменений может неблагоприятно сказаться на внешнем виде, компоновке и функционировании сборки в целом. Помимо этого, ограничения используются для более точной связи геометрических элементов, позволяя создавать между ними логические взаимосвязи.

Геометрические ограничения - ограничения, которые определяют расположение элементов эскиза относительно друг друга и относительно ранее созданной геометрии текущей модели.

Размерные ограничения – ограничения, задающие расстояние между двумя элементами. В зависимости от типа участвующих в операции геометрических элементов это расстояние может быть линейным, угловым или радиальным.

При работе со сборкой также необходимо накладывать ограничения на детали для определения взаимного расположения компонентов. Ниже приведен список некоторых ограничений:

Ограничение Coincidence (Совпадение) обеспечивает соосность, компланарность или совпадение точек.

Ограничение Contact (Контакт) соединяет две плоскости или грани.

Ограничение Offset (Смещение) определяет расстояние между двумя элементами.

Ограничение угла Angle (Угол) определяет угол между элементами компонентов. [2]

Расчеты прочности изделия

Статические расчеты конструкций на прочность являются особо важными в машиностроительном проектировании. Часто в машиностроительном проектировании появляется необходимость оценить напряженное состояние отдельных деталей или конструкции в целом [3-4].

Обычно при проверочном расчете изделия на прочность нас интересует распределение составляющих напряжений по объему элементов конструкции и максимально возможные значения напряжения компонентов в материале (рис. 1).

Большинство современных САПР позволяют конструктору на этапе проектирования изделия решить обе эти задачи, обеспечив, таким образом, высокие механические характеристики будущего изделия.

Прочностные расчеты содержат:

- Анализ потери устойчивости конструкций
- Расчет напряженно-деформированного состояния конструкций при нагрузках
- Расчет ветровых и волновых нагрузок на конструкции и т. д.

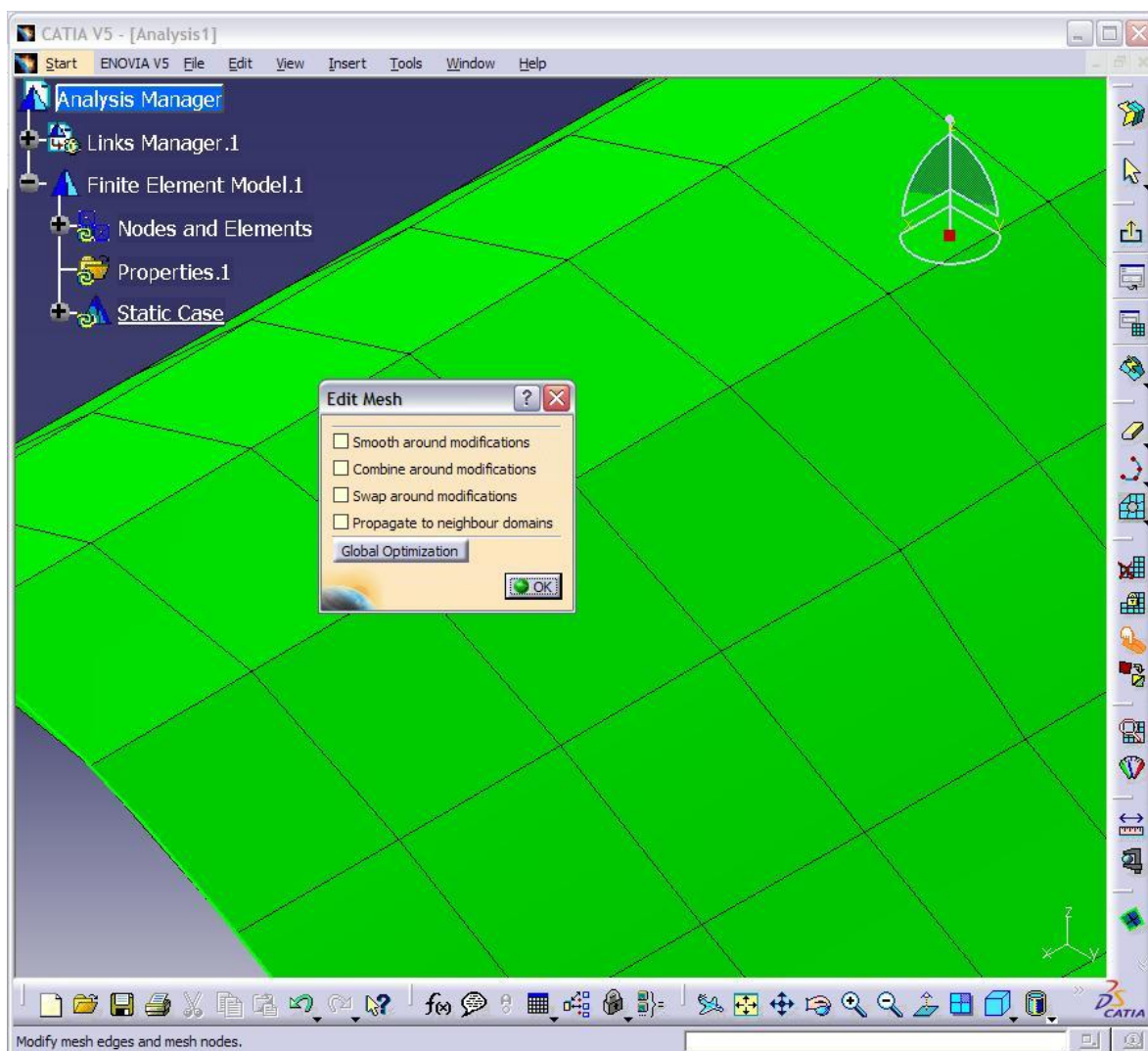


Рисунок 1 – Прочностные и иные расчеты в Catia v5

Термодинамические расчеты

Изделия, требующие оценки теплового режима, весьма разнообразны — это и различные электрические устройства, выделяющие тепловую энергию, и специализированные термические установки, и тепловые двигатели.

Как правило, перед разработчиком ставится задача оценки температурных режимов деталей изделия или изделия целиком. То есть, необходимо определить температуры, возникающие в изделии при воздействии приложенных к системе источников тепловой энергии (рис. 2).

Тепловые расчеты включают в себя [3-4]:

- Анализ термических напряжений в конструкциях (моделирование напряжений в случае пожаров или циклических тепловых нагрузок);
- Термогидравлические расчеты трубопроводов (моделирование протекания потоков в сложных системах труб с учетом тепловых потоков на стенах);
- Моделирование работы систем вентиляции (полная картина воздушных потоков, создаваемых вентиляцией) и т.д.

Однако следует отметить, что наличие тепловых моделей является довольно сложной функцией САПР.

Не все САПР поддерживают возможности физического моделирования тепловых процессов. Можно назвать, для примера, САПР, поддерживающие такую возможность, такие, как:

- CATIA;
- AVEVA;
- SolidWorks и т.д.

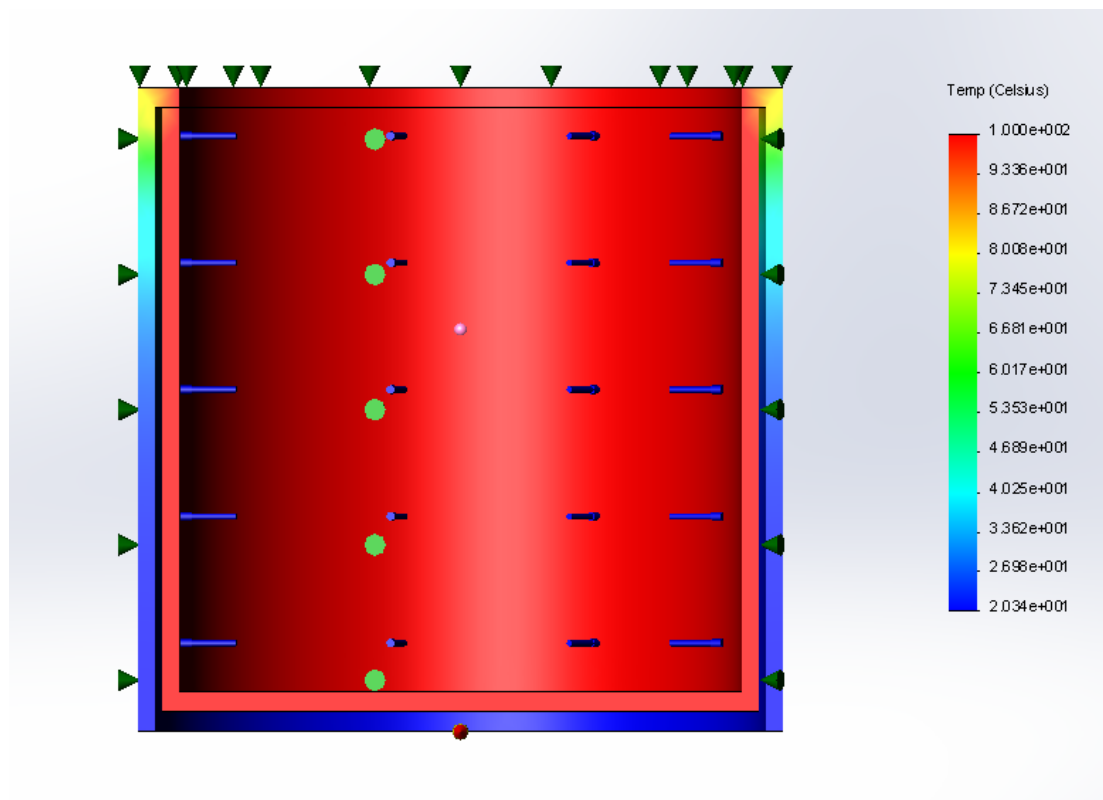


Рисунок 2 – Симуляция нагрева тела в SolidWorks Simulation

Выводы

Проделанный анализ позволяет сказать, что для полноценного представления о работе проекта в реальной жизни нам необходимо смоделировать ее на компьютере во избежание непредвиденных проблем. Также хотелось бы отметить, что в данной работе представлены далеко не все возможности комплекса САЕ.

Литература

1. САПР и Графика. Оживляя механизмы. [Электронный ресурс]. – Режим доступа URL: <https://sapr.ru/article/24477>
2. Создание, ограничения и движение в CATIA v5: Статья/ Власюк Д. А., 2017
3. CFD group [Электронный ресурс]. – Режим доступа URL: <http://www.cfdgroup.ru/>
4. Форум CAD/CAM/CAE/PLM [Электронный ресурс]. – Режим доступа URL: <http://ccsr3d.ru/topic/69265-симуляция-нагрева-тела/>

УДК 004.4'23

ИССЛЕДОВАНИЕ API САПР CATIA V5 И ПЕРСПЕКТИВЫ РАЗВИТИЯ СОЗДАНИЯ ПРИЛОЖЕНИЙ В НЕЙ

Бондаренко Е.С., Григорьев А.В.

Донецкий национальный технический университет

Кафедра «Программной инженерии»

E-mail: katusha.bondarenko@yandex.ua

Аннотация:

Бондаренко Е.С., Григорьев А.В. Исследование API САПР CATIA V5 и перспективы развития создания приложений в ней. В работе проведен сравнительный анализ геометрических ядер САПР. Выделены преимущества использования языка программирования Python для создания приложений и рассмотрены перспективы развития API САПР.

Annotation:

Bondarenko E.S., Grigoriev A.V. Research of API CATIA V5 and the perspectives for the development of creating applications in it. A comparative analysis of geometric CAD cores was performed. The advantages of using the Python programming language to create applications are highlighted and the prospects for the development of the CAD API are considered.

Ключевые слова: API, Python, разработка приложений, интерфейс, язык программирования.

Введение

Сейчас создано множество универсальных систем автоматизированного проектирования для повышения эффективности труда инженеров и облегчения процесса производства. Они обладают мощной функциональностью моделирования и широкими возможностями встроенных инструментальных средств. Но даже этих решений для выполнения конкретных производственных задач иногда бывает недостаточно.

Тогда на помощь специалистам приходит интерфейс прикладного программирования (API), то есть некоторый набор готовых классов, процедур, функций, структур и констант, созданный для упрощения написания пользовательских программных модулей, интегрированных с САПР. Наличие в САД-системах API позволяет осуществлять тонкую настройку системы проектирования и помогает сократить время разработки дополнительных программных компонентов к уже имеющейся на предприятии САПР.

Однако ни одна из имеющихся систем автоматизированного проектирования не обладает полным набором функциональных возможностей для ее полноценного использования специалистами.

Целью работы является исследование возможности использования языка Python в среде САПР Catia V5 для создания дополнительных модулей и приложений. Для достижения поставленной цели были выявлены следующие задачи:

- изучить особенности API в Catia V5;
- провести анализ САПР и используемых ими геометрических ядер;
- выделить преимущества и недостатки языка Python;
- привести пример создания макроса для редактирования имен в VBA;
- систематизировать приложения, которые можно написать в Catia и привести пример операций на языке Python для построения эскизов в Catia;

- сделать вывод по полученным результатам и предложить перспективы развития современных САПР.

API в CATIA V5 и его особенности

Зачем API в САПР?

1. Автоматизация чертежей;
2. Упрощение создания и редактирования чертежей.

API дает возможность полностью контролировать исходный код, оперативно исправлять ошибки и наращивать функционал. Это в своих системах используют разработчики [CATIA](#), [NX](#), [Creo Elements/Pro](#), [Solid Edge](#), [Autodesk Inventor](#), [think3](#), [КОМПАС-3D](#).

Новая разработка, получившая кодовое имя CNext, в 1998 году стала известной всему миру под названием CATIA V5. Она имеет открытую архитектуру — в нее можно вставить собственные наработки и программы (например, тепловой расчет и т.д.), функции, необходимые для расширения возможностей специализированных модулей. Использование программирования на языках VB CATIA, Java, C++, VBA, Basic Script позволяет автоматизировать весь процесс проектирования. API для CATIA называется «САА» - сокращение от Component Application Architecture. Этот API написан на C++. CATIA имеет другие API, основанные на Microsoft Visual Basic и Java, но они не такие полнофункциональные и не такие гибкие, как САА C++ API. Вместо устаревшего Fortran в качестве языка программирования был выбран C++, а геометрическое ядро новой системы получило название [CGM](#) – CNext/CATIA Geometric Modeler.

Особенности работы с интерфейсом прикладного программирования прописываются в технической документации. Здесь же указывается, какие значения необходимо подавать на вход конкретной функции, чтобы получить на ее выходе корректные значения, согласно предназначению данной функции. Таким образом, API позволяет абстрагироваться от реализации отдельных программных блоков при разработке приложений. Наиболее явно эта концепция раскрывается на примере черного ящика (рис. 1).

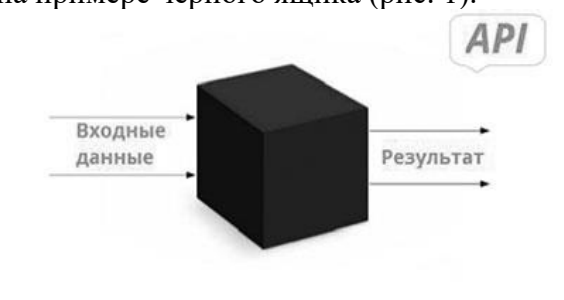


Рисунок 1 – «Черный ящик»

Физически API-функции представляются в виде отдельного программного модуля, который динамически подключается извне к основному проекту в формате DLL-библиотеки.

В целом, приложения для CATIA можно писать в виде:

1. Макросов *.catvba, скриптов *.CATScript;
2. Автономных приложений, скомпилированных в *.exe-файл. Можно использовать любой язык программирования, например VB6, C#, Delphi. При этом взаимодействие с CATIA происходит как с COM-объектом;
3. В виде приложений, плагинов, используя САА-RADE - на C++.

Самый простой вариант - это приложения в виде макросов *.catvba. Создать макрос в CATIA можно так: меню Tools / Macro / Macros... Далее на кнопку Macro libraries... / выбрать Library type - VBA projects / Create new library. Потом в этом макросе *.catvba можно добавить модули - кнопкой Create в окне Macros, а также в редакторе Microsoft Visual Basic - модули, формы, классы.

Пример создания макроса для редактирования имен Part'ов.

```
Sub CATMain()
  Dim sel As Selection
  Dim i As Long
  Dim iProd As Product
  Set sel = CATIA.ActiveDocument.Selection
  If sel.Count > 0 Then
    For i = 1 To sel.Count
      If sel.Item(i).Type = "Product" Then
        Set iProd = sel.Item(i).Value
        MsgBox iProd.Name
      End If
    Next
  End If
End Sub.
```

Пакет геометрического моделирования – набор библиотек с программным интерфейсом (API), с помощью которого можно пользоваться функциями геометрического (например, твердотельного) моделирования. Многие ведущие CAD-системы (CATIA, Pro/E, NX) построены на основе собственных геометрических ядер (CGM, GRANITE, Parasolid соответственно), тогда как другие (SolidWorks, T-FLEX, ADEM и прочие) построены на основе лицензированных геометрических ядер. Популярными коммерческими ядрами являются Parasolid (от компании UGS), ACIS (Spatial Corp. – дочерней компанией Dassault Systems), GRANITE (PTC). Свободно распространяется в открытом коде ядро Open CASCADE. В таблице для сравнения приведены системы проектирования с описанием используемых ими ядер.

Ядро	Доступно с	Разработчик	САПР, основанные на ядре
ROMULUS	1982	Shape Data	HP ME30, CAM-I A
Designbase	1986	Ricoh	CADRA, Helix, ICAD, GMSWorks
ACIS	1989	Spatial (Dassault Systemes)	ADEM, Alibre Design, Bricscad, Creo Elements/Direct (частично), IRONCAD, KeyCreator, TurboCAD, SpaceClaim
Parasolid	1989	Siemens Software	PLM IRONCAD, KeyCreator, MicroStation, NX, Solid Edge, SolidWorks, T-FLEX
SMLib	1998	Solid Modeling Solutions	?
Open CASCADE	1999	OPEN CASCADE (Areva)	CAD-Schroer, FreeCAD
GRANITE	2001	PTC	Creo Elements/Pro (Pro/Engineer)
SOLIDS++	2004	IntegrityWare	Rhino (частично)
CGM	2010	Dassault Systemes	CATIA V5, V6, SolidWorks V6

Использование Python для работы с Catia

CATIA изначально использует Visual Basic для приложений (VBA), Однако функциональность VBA ограничена, особенно когда требуется анализ данных и построение графиков. Python - это гораздо более мощный инженерный язык, который работает со многими программами, такими как Microsoft Excel и многими другими. Этот язык программирования имеет ряд преимуществ над другими, используемыми в САПР:

1. интерпретатор Python реализован практически на всех платформах и операционных системах. Первым таким языком был С, однако его типы данных на разных машинах могли занимать разное количество памяти и это служило препятствием при написании переносимой программы. Python же таким недостатком не обладает.
2. расширяемость языка. Это означает, что имеется возможность совершенствования языка любыми программистами. Интерпретатор написан на С и исходный код доступен для любых манипуляций. Написав на С свои дополнения к Python и скомпилировав программу, можно получить "расширенный" интерпретатор с новыми возможностями.
3. наличие большого числа подключаемых к программе модулей, обеспечивающих различные дополнительные возможности:
 - Numerical Python - расширенные математические возможности, такие как манипуляции с целыми векторами и матрицами;
 - Tkinter - построение приложений с использованием графического интерфейса (GUI) на основе широко распространенного на X-Windows Tk-интерфейса;
 - OpenGL - использование обширной библиотеки графического моделирования двух- и трехмерных объектов Open Graphics Library фирмы Silicon Graphics Inc. Данный стандарт поддерживается, в том числе, в таких распространенных операционных системах как Microsoft Windows 95 OSR 2, 98 и Windows NT 4.0.

Однако недостатком Python является сравнительно невысокая скорость выполнения Python-программы, что обусловлено ее интерпретируемостью.

Таким образом, язык Python может продвигаться в сфере программирования в САПР и стать одним из наиболее широко используемых. Этот язык с открытым исходным кодом и может быть свободно загружен без лицензионных ограничений.

Самая первая операция python - импортировать используемые модули. Затем ядро Python пытается подключиться к приложению CATIA, работающему в Windows.

Импорта питонного модули

импорт ОС из win32com.client импорта диспетчерских

Подключения к окнам COM

CATIA = диспетчерская («CATIA.Application»)

опции CATIA видимости

CATIA . Visible = True

Далее, Part's Sketch создаются в инструментальной среде CATIA Part Design.

Создать пустую деталь

partDocument1 = CATIA . Документы . Добавить («Часть»)

Создать точку, на которую ссылаются плоскость и эскиз

Xcoord = 100

Ycoord = 100

Zcoord = 100

NewPoint = CATIA . ActiveDocument . Часть . HybridShapeFactory AddNewPointCoord (Xcoord , Ycoord , Zcoord)

Mainbody = CATIA . ActiveDocument . Часть . MainBody

Mainbody . InsertHybridShape (NewPoint)

Создать базовую плоскость для эскиза

AxisXY = CATIA . ActiveDocument . Часть . OriginElements . PlaneXY

Referenceplane = CATIA . ActiveDocument . Часть . CreateReferenceFromObject (AxisXY)

```
Контрольная точка = CATIA . ActiveDocument . Часть . CreateReferenceFromObject (
NewPoint )
NewPlane = CATIA . ActiveDocument. Часть .HybridShapeFactory . AddNewPlaneOffsetPt (
эталонная плоскость, эталонная точка )
. InsertHybridShape ( NewPlane )

# создать эскизные
наброски1 = CATIA . ActiveDocument . Часть . Тел . Предмет ( "PartBody" ) . Эскизы
reference1 = partDocument1 . часть . OriginElements . PlaneXY
NewSketch = sketches1 . Добавить ( reference1 )
CATIA . ActiveDocument . Часть . InWorkObject = NewSketch

# Нарисовать эскиз
NewSketch . OpenEdition ()
# Начало (H, V) Конец (H, V)
NewLine1 = NewSketch . Factory2D . CreateLine ( 0 , 0 , 0 , 50 )
NewLine2 = NewSketch . Factory2D . CreateLine ( 0 , 50 , 50 , 50 )
NewLine3 = NewSketch . Factory2D . CreateLine ( 50 , 50 , 50 , 55 )
NewLine4 = NewSketch . Factory2D . CreateLine ( 50 , 55 , 150 , 55 )
NewLine5 = NewSketch . Factory2D . CreateLine ( 150 , 55 , 150 , - 5 )
NewLine6 = NewSketch . Factory2D . CreateLine ( 50 , - 5 , 50 , 0 )
NewLine7 = NewSketch . Factory2D . CreateLine ( 50 , - 5 , 75 , - 5 )
NewLine8 = NewSketch . Factory2D . CreateLine ( 150 , - 5 , 125 , - 5 )
NewLine9 = NewSketch . Factory2D . CreateLine ( 125 , - 5 , 125 , 25 )
NewLine0 = NewSketch . Factory2D . CreateLine ( 125 , 25 , 75 , 25 )
NewLine11 = NewSketch . Factory2D . CreateLine ( 75 , 25 , 75 , - 5 )
NewLine12 = NewSketch . Factory2D . CreateLine ( 50 , 0 , 0 , 0 )
NewSketch . CloseEdition ()
```

Метод, использующий CATIA CAD и язык программирования Python, представлен как метод для улучшения моделирования и анализа путем автоматизации шагов, необходимых для получения результатов моделирования.

Выводы

В работе исследованы возможности и инструменты создания API в Catia V5. Наиболее эффективным инструментом создания API является Python, т.к. в процессе разработки программного обеспечения он позволяет быстрее создавать CAD, CAM и CAE-системы. Рассмотрены примеры использования языка Python в среде САПР Catia V5 для создания дополнительных модулей и приложений.

Литература

4. Д. Зитен . Программирование макросов CATIA V5 с помощью скрипта Visual Basic, McGraw-Hill, 2013.
5. В.Н. Малюх. Введение в современные САПР, М.: ДМК Пресс, 2010.
6. Обзор популярных систем автоматизированного проектирования (cad) [Электронный ресурс]. Режим доступа: <https://www.pointcad.ru/novosti/obzor-sistem-avtomatizirovannogo-proektirovaniya>. Дата последнего посещения: 05.05.2019.

УДК 004.94

ПРОБЛЕМАТИКА МОДЕЛИРОВАНИЯ КРАШ-ТЕСТОВ ДЛЯ МОДЕЛЕЙ МОТОЦИКЛОВ

Шевелёв И.А., Брюхова Д.В., Григорьев А.В.

Донецкий национальный технический университет

Кафедра «Программной инженерии»

E-mail: bryuhova2000@yandex.ru

Аннотация:

Шевелёв И.А., Брюхова Д.В., Григорьев А.В. Проблематика моделирования краш-тестов для моделей мотоциклов. В статье описываются возможности виртуального тестирования моделей автомобилей, а также аргументируется необходимость создания краш-тестов и других испытаний для мотоциклов.

Annotation:

Shevelev, I.A., Bryukhova, D.V., Grigoriev, A.V. The problem of crash test modeling for motorcycle models. The article describes the possibilities of virtual testing of car models, and also argues the need to create crash tests and other tests for motorcycles.

Введение

В нашу эпоху стремительного развития машиностроения актуальной проблемой является большие финансовые затраты на проведение различных краш-тестов. При появлении возможности точных виртуальных краш-тестов затраты заметно уменьшатся, а количество проведенных испытаний возрастет, что даст возможность повысить безопасность людей при авариях.

Краш тестам можно подвергать модели любого транспорта – легковые и грузовые автомобили, мотоциклы, военную технику, авиационную технику, трубопроводный транспорт и т.д.

Каждый вид транспорта предполагает свою специфику и – свои требования к средствам моделирования.

Цель данной статьи – проанализировать типичные возможности существующих систем проведения модельных краш-тестов, а также определить специфику модельных краш-тестов для такого вида транспорта, как мотоциклы.

Анализ возможностей инновационной платформы HyperWorks как средства моделирования краш-тестов для моделей автомобилей

Данная платформа разработана с целью оптимизации проектирования и управления данными расчетов и данными о характеристиках модели, а также автоматизации производственных процессов. Являясь одной из ведущих CAE-систем для решения задач промышленности, программа HyperWorks предоставляет четко интегрированный пакет лучших в своем классе инструментов для моделирования, анализа, оптимизации, визуализации и обработки результатов исследований, а также управления данными по характеристикам проектируемого изделия [1].

Для создания элемента в среде HyperCrash необходимо импортировать модель транспорта, которая разграничена по деталям, где каждая деталь имеет свои признаки, прочность, структуру.

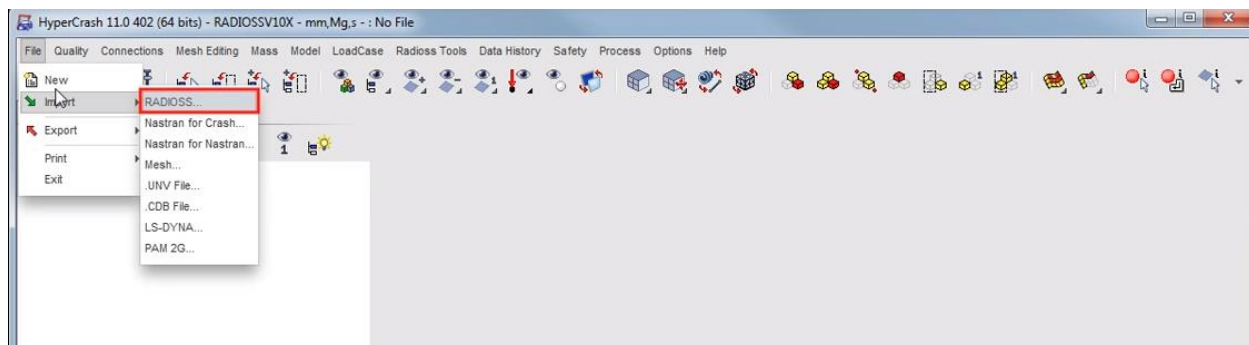


Рисунок 1 – Импорт RADIOSS модели

Каждая деталь представлена в виде отдельной модели в среде HyperCrash. Для визуального отличия каждая деталь обладает своим цветом.

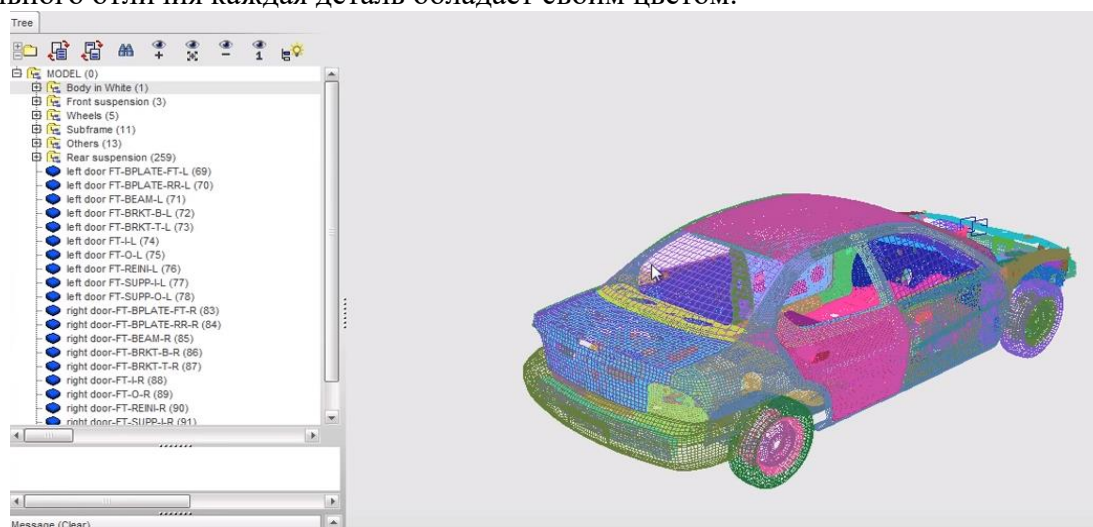


Рисунок 2 – Модель машины в среде HyperCrash

Детали соединены между собой креплениями, которые:

1. Представлены в виде отдельных моделей (например, крыло, которое прикручено болтами в специальных креплениях, где каждый элемент представляет собой отдельную деталь);
2. Представлены в виде связи в HyperCrash между элементами (рис. 3).

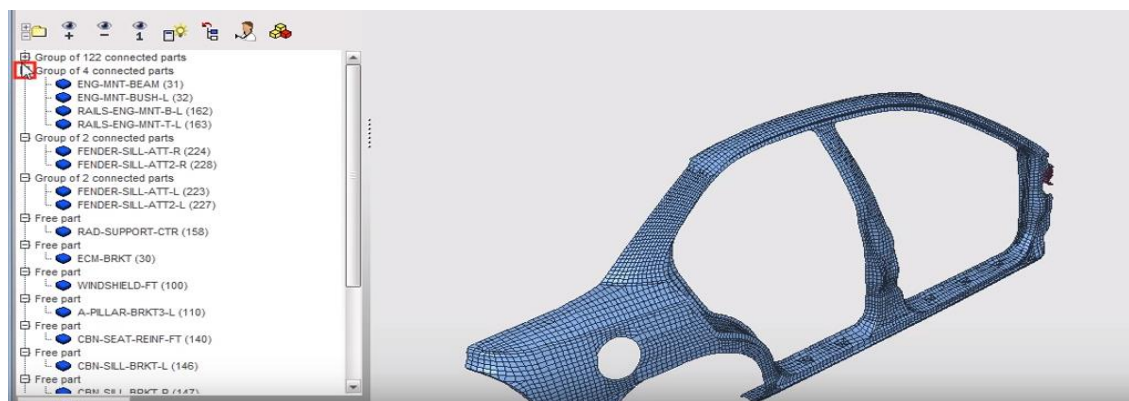


Рисунок 3 – Группа связанных элементов

Модуль можно использовать, не имея специальных знаний в области расчетов, методом конечных элементов. Схема его использования сведена к простейшим процедурам:

- присвоения материала;
- указания мест и условий закрепления;

- указания мест и условий нагружения[2].

После автоматической генерации сетки конечных элементов, выбора оптимального расчетного метода и собственно расчета, модуль представляет результаты в наглядной и легко понимаемой форме. На основании этих результатов конструктор может легко сделать вывод о том, как нужно изменить деталь, чтобы оптимизировать ее механические характеристики.

Если при импорте элементов модели деталей теряют программное соединение, необходимо перейти во вкладку Connection -> Spotweld -> Create. Затем необходимо указать поверхности, к которым будет применяться закрепление.

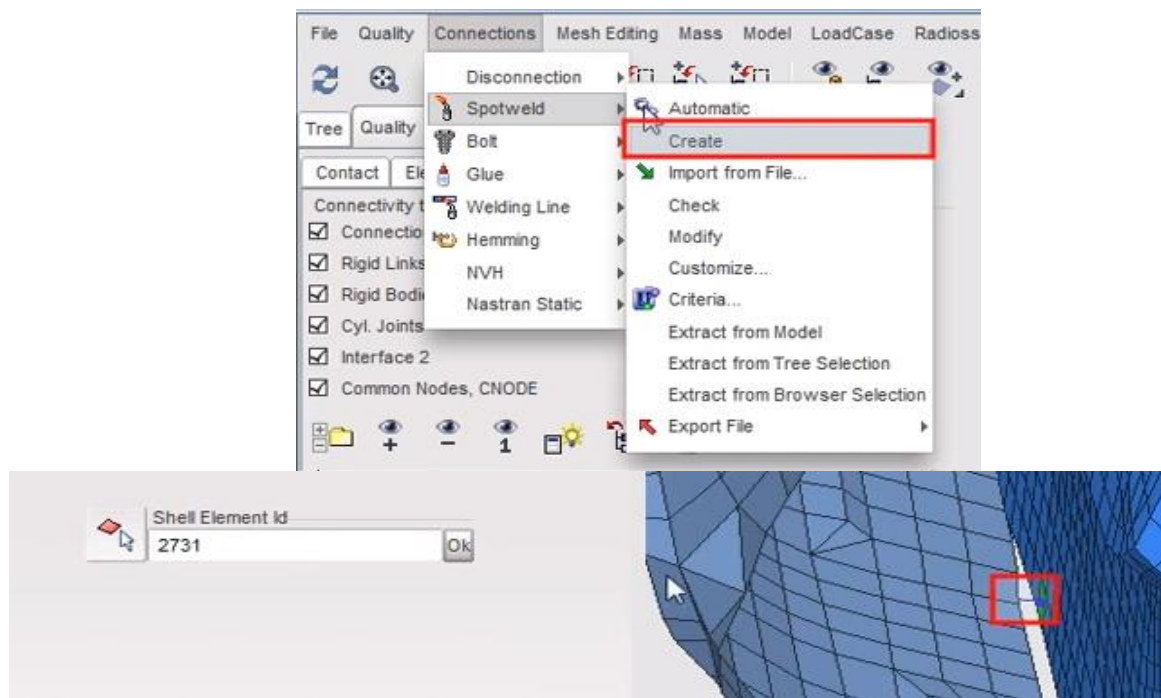


Рисунок 5 – Соединение элементов по ID

При запуске симуляции все элементы не деформированы, отображается шкала силы деформации.

Для проведения прочностных расчетов необходимо:

- указать места и условия закрепления детали;
- указать места и условия приложения силы;
- запустить вычисление нагрузок при заданных параметрах.

Выводы

В наше время в автокатастрофах гибнет большое количество людей, поэтому проблема безопасности стоит на первом месте у производителей автомобилей и мотоциклов. Именно из-за высоких приоритетов безопасности количество смертей в ДТП заметно сокращается.

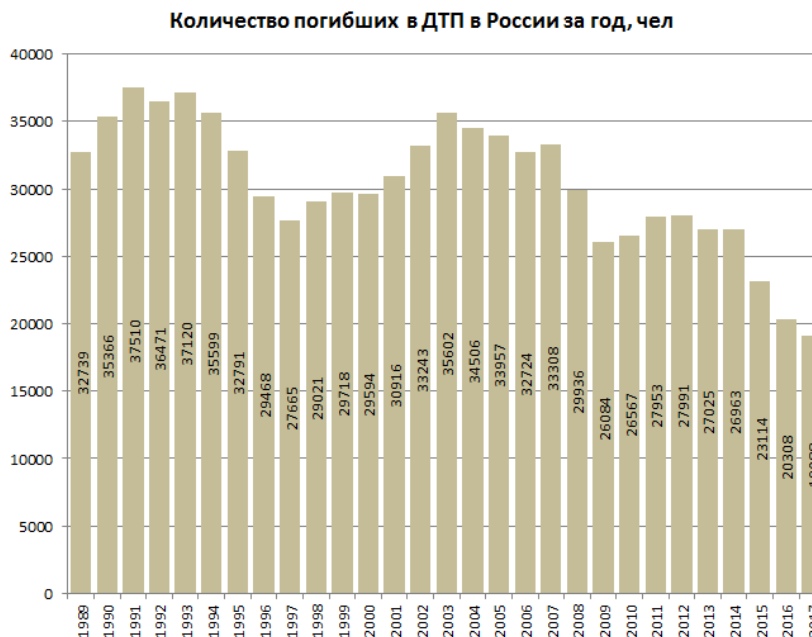


Рисунок 6 – Диаграмма количества погибших в ДТП

Чтобы сделать мотоциклы более безопасным видом транспорта и тратить на это меньше ресурсов, необходимо иметь возможность проводить виртуальные краш-тесты с моделями, как показывалось ранее с автомобилями. Требуется создать дополнительный модуль для мотоциклов с испытанием на прочность модели, оптимизацией конструкции мотоцикла для увеличения безопасности при столкновении с автомобилем, смоделировать движение мотоцикла по различным дорожным покрытиям, а также при любых погодных условиях.

На данный момент результаты существующих модулей CAE для испытаний моделей автомобилей недостаточно точны. Чтобы решить эту проблему, требуется уделить достаточное количество времени для усовершенствования этих модулей, добавления больших количеств испытаний и моделирования различных ситуаций на дорогах.

Краш-тесты как для автомобилей, так и для мотоциклов являются неотъемлемой частью их разработки и производства. На основании полученных данных от испытания производитель, если это требуется, вносит изменения в конструкцию, направленные на повышение безопасности людей при авариях. Краш-тесты дали огромный толчок для появления таких на сегодня общепринятых вещей, как ремни и подушки безопасности, зоны электропоглощения кузова и многое другое. Реальные краш-тесты позволяют с высокой точностью определить, что происходит с пассажирами и водителем при аварии и какие перегрузки они испытывают. Так как результаты испытаний влияют на модель, нужно виртуальные краш-тесты довести до максимальной точности и минимизировать вероятность погрешности и ошибки в них.

Литература

1. HyperWorks [Электронный ресурс]. – Режим доступа URL: <http://www.hyperworks.compmechlab.ru/article/hyperworks-student-version>
2. HyperWorks [Электронный ресурс]. – Режим доступа URL: <https://www.rodnik.ru/catalog/sistemy-inzhenernogo-analiza/altair-hyperworks/>
3. CATIA [Электронный ресурс]. – Режим доступа URL: <https://sapr.ru/article/7101>

СЕКЦИЯ 4

ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ И
АВТОМАТИЗИРОВАННЫЕ
СИСТЕМЫ УПРАВЛЕНИЯ

ИУС
МКМ 2019

УДК 007.51+004.021

ДВУХУРОВНЕВЫЙ ГЕНЕТИЧЕСКИЙ АЛГОРИТМ ОПТИМИЗАЦИИ ГРУЗОПЕРЕВОЗОК

Бражник А.В.*, Секирин А.И.*, Валуева О.С.**

Донецкий национальный технический университет

*кафедра автоматизированных систем управления,

**кафедра автоматики и телекоммуникаций

E-mail: alexeybrazhnik1996@gmail.com

Аннотация:

Бражник А.В., Секирин А.И., Валуева О.С. Двухуровневый генетический алгоритм оптимизации грузоперевозок. В данной статье описаны основные особенности задачи оптимизации перевозок, частью которой является задача коммивояжера. Приведена математическая постановка задачи. Приведено описание разработанного алгоритма оптимизации грузоперевозок. Приведен предполагаемый вариант взаимодействия разрабатываемого алгоритма с другими алгоритмами оптимизации и подсистемами предприятия.

Annotation:

Brazhnik A.V., Sekirin A.I., Valueva O.S. The two-level genetic algorithm to optimize cargo transportation. This article describes the main features of the optimization of transportation, part of which is the traveling salesman problem. The mathematical formulation of the problem is given. The description of the developed cargo optimization algorithm is given. The proposed version of the interaction of the developed algorithm with other optimization algorithms and company subsystems is given.

Введение

Текущий уровень развития технологий обуславливает повсеместное применение грузоперевозок на предприятиях, осуществляющих различные виды деятельности. В текущее время наиболее распространенными являются автомобильные грузоперевозки, по причине как относительной дешевизны, так и возможности транспортировки груза напрямую из точки погрузки к адресату или пункту выгрузки благодаря достаточно разветвленной сети автомобильных дорог. Однако не все предприятия способны иметь в своем составе грузовой автопарк и сопутствующую инфраструктуру, в связи с чем успешно функционируют специализированные предприятия, называемые транспортными компаниями.

Транспортная компания осуществляет перевозку грузов между различными пунктами для различных заказчиков. Для осуществления такой деятельности в компании имеется соответствующая инфраструктура и ведется учет работы с использованием программных средств. На сегодняшний день развитие транспортных компаний и увеличение их количества происходит высокими темпами, в связи с чем усиливается конкуренция и появляется необходимость оптимизации работы транспортной компании с целью уменьшения затрат и увеличения прибыли [1].

Оптимизация работы транспортной компании подразумевает сокращение расходов на перевозку груза, требует длительных арифметических вычислений, поэтому производится с использованием специальных программных средств, называемых логистическими системами. Каждая из логистических систем использует свои собственные алгоритмы для оптимизации работы, причем эти алгоритмы скрыты от конечного пользователя системы. В текущее время существует большое количество логистических систем, каждая из которых

предоставляет свой, отличный от других, функционал, учитывая при этом большое количество факторов [2]. Однако быстрое развитие транспортных компаний обуславливает постоянное изменение условий их работы, появление новых факторов, которые не в полной мере учитываются существующими логистическими системами.

Алгоритмы существующих логистических систем зачастую не адаптированы к постоянно изменяющимся условиям работы транспортной компании, в связи с чем все более актуальной становится необходимость модификации этих алгоритмов а также разработки новых алгоритмов, учитывающих все факторы, влияющие на работу транспортных компаний в настоящее время [3].

Постановка задачи

Рассматриваемая задача оптимизации работы транспортной компании является частным случаем задачи коммивояжера, в котором присутствует разбиение на несколько начальных пунктов, в данном случае это пункты загрузки транспортных средств. Эти пункты расположены на различном удалении от пунктов выгрузки транспортных средств, таким образом, для каждого из пунктов загрузки необходимо составить свою собственную матрицу стоимости перевозок.

Целевая функция для задачи определяется по формуле 1.

$$\sum_{i=1}^N C_i L_i \rightarrow \min, \quad (1)$$

где C_i – стоимость перевозки груза i -м автомобилем на единицу расстояния, L_i – суммарная длина маршрута, по которому едет i -й автомобиль, N – количество автомобилей.

При этом необходимо учесть следующие ограничения, определяемые формулой 2.

$$\left\{ \begin{array}{l} 1. \sum_{j=1}^N M_j \leq MA_i, \\ 2. \sum_{j=1}^N V_j \leq VA_i, \\ 3. T_j \leq T_{jk}, \\ 4. K_{отпр} > 1, \end{array} \right. \quad (2)$$

где N – количество грузов в одном автомобиле, M_j – масса j -го груза в автомобиле, MA_i – грузоподъемность i -го автомобиля, V_j – объем j -го груза в автомобиле, VA_i – объем кузова i -го автомобиля, T_j – фактическое время разгрузки j -го груза, T_{jk} – критическое время разгрузки j -го груза, $K_{отпр}$ – количество точек отправления автомобилей.

При этом ограничение 1 определяет, что суммарная масса груза, перевозимого одним автомобилем, не может быть больше его грузоподъемности. Ограничение 2 определяет, что суммарный объем груза, перевозимого одним автомобилем, не может быть больше его объема. Ограничение 3 определяет, что время доставки груза в точку выгрузки должно быть не позднее некоторого $T_{критического}$, которое задано в исходных данных. Ограничение 4 определяет, что пунктов отправки автомобилей несколько (более 1-го).

Также необходимо учесть следующие дополнительные ограничения:

1. грузоподъемность автомобилей различна
2. каждый груз может быть перевезен только одним автомобилем
3. пункты выгрузки различны для каждого груза

Критерием оптимальности маршрута согласно формуле 1 является минимальная суммарная стоимость перевозки всех грузов транспортными средствами.

Описание разработанного алгоритма оптимизации грузоперевозок

Поставленную задачу оптимизации грузоперевозок предлагается решать с использованием двухуровневого генетического алгоритма, в котором на первом уровне оптимизируется загрузка транспортного средства с учетом ограничений на параметры транспортного средства, а на втором уровне алгоритма оптимизируется маршрут перевозки между пунктами разгрузки с учетом ограничений на время доставки груза и различные пункты отправки.[4]

При этом на первом уровне генетического алгоритма критерием оптимальности является оптимальное распределение груза между выбранными транспортными средствами, а на втором уровне критерием оптимальности является минимальная длина пути. Также на первом шаге алгоритма проверяются следующие ограничения: ограничение по массе, по объему, и то, что груз неделим и может быть перевезен только в одном автомобиле целиком. Для учета ограничений на несколько пунктов отправки используются различные матрицы расстояний для каждого из пунктов отправки.

Укрупненная схема работы двухуровневого генетического алгоритма для оптимизации грузоперевозок приведена на рисунке 1.

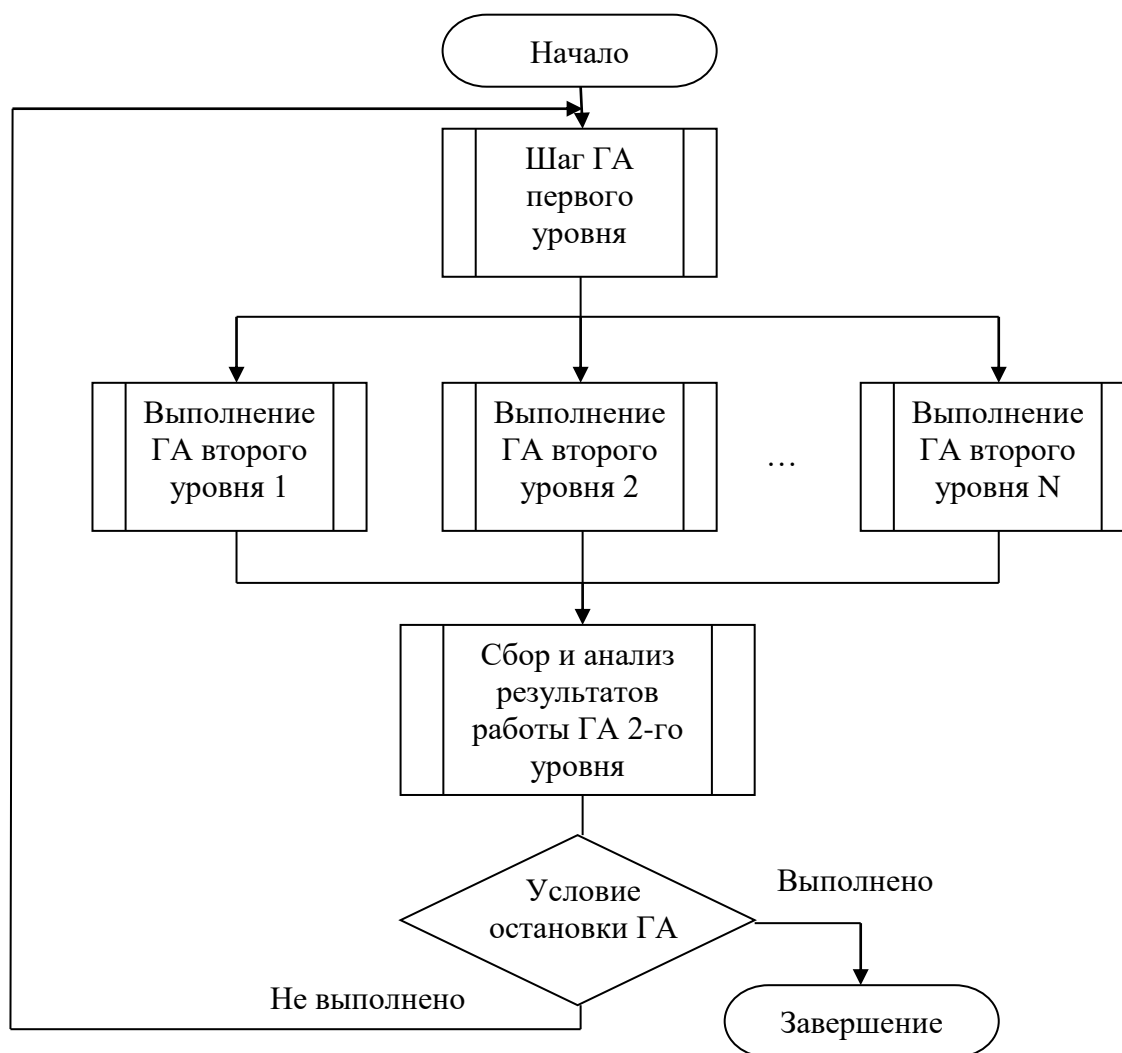


Рисунок 1. Укрупненная схема работы генетического алгоритма

Как видно из схемы, после выполнения каждого шага первого уровня генетического алгоритма выполняется запуск N генетических алгоритмов второго уровня, после чего выполняется сбор и анализ результатов работы генетических алгоритмов второго уровня, а затем, в случае если не выполнено условие остановки алгоритма, выполняется переход на следующий шаг алгоритма. Количество генетических алгоритмов второго уровня равно мощности популяции генетического алгоритма первого уровня. Как видно из схемы, генетические алгоритмы второго уровня могут быть выполнены параллельно при наличии достаточного количества ресурсов вычислительной системы. В работе первого уровня разрабатываемого генетического алгоритма используется хромосома, структура которой приведена в таблице 1.

Таблица 1. Структура хромосомы генетического алгоритма первого уровня

T1	Rп1	C11	C12	...	C1N1
T2	Rп2	C21	C22	...	C2N2
...
TM	RпM	CM1	CM2	...	CMNM

где M – количество транспортных средств, Ni – количество грузов в i – м транспортном средстве, Rпi – пункт погрузки для i – го транспортного средства, Cij – j-й груз в i – м транспортном средстве.

Фитнесс-функция для первого этапа генетического алгоритма представляется суммарной стоимостью перевозки загруженных грузов всеми загруженными автомобилями и рассчитывается по формуле 3.

$$F = \sum_{i=1}^M \left(\sum_{j=1}^{Ni} A_{kj-1kj} C_i \right) \rightarrow \min , \quad (3)$$

где F – значение фитнесс-функции, i – номер транспортного средства, M – количество транспортных средств в хромосоме (строк в таблице), j – номер столбца в таблице, Ni – количество грузов в i-м транспортном средстве, Akj-1kj – расстояние между пунктами Kj-1 и Kj из матрицы расстояний, Ci – стоимость перевозки груза i-м транспортным средством на единицу расстояния.

На основании значений фитнесс-функции производится отбор особей в популяции для выполнения кроссинговера. На первом поколении генетического алгоритма оценка фитнесс-функции особей производится по формуле 1 на основании такого порядка, который был сформирован при формировании начальной популяции. В качестве стратегии для отбора особей в популяции предполагается выбрать ранжирование либо случайный отбор.

Полученные в ходе отбора группы особей (пары) с определенной вероятностью принимают участие в кроссинговере. Структура хромосомы в виде таблицы определяет типы кроссинговера. Наиболее вероятными типами кроссинговера являются обмен строками в таблицах, обмен подстроками в таблицах и комбинированный обмен строками и подстроками.

Кроссинговер «обмен строками в таблицах» представляет собой случайный выбор строк из первой хромосомы-родителя и взаимную замену строк с такими же кодами транспортных средств из второй хромосомы-родителя. При таком типе кроссинговера может потребоваться корректировка хромосом после замены, поскольку один груз может быть загружен в два транспортных средства в пределах одного решения, тем самым может быть нарушено ограничение на перевозку груза одним транспортным средством. В таком случае целесообразнее с точки зрения временной эффективности работы алгоритма выполнять проверку корректности обмена до выполнения операции кроссинговера.

Пример кроссинговера «обмен строками в таблицах» приведен на рисунке 2. Стрелкой показан обмен строк с кодами транспортных средств равными 4, также обмениваемые строки выделены более толстой границей строки. Для наглядности в каждую

из таблиц, представляющих хромосому, добавлена строка заголовка, показывающая, примеры каких значений находятся в каждом столбце хромосомы. Следует отметить, что в данном примере показан случай, при котором не возникла коллизия при обмене строками.

Хромосома-родитель 1

Код транс. средства	Код п. отправки	Код груза 1	Код груза 2	Код груза 3
1	1	13	10	4
4	2	7	9	
5	1	3	15	6

Хромосома-родитель 2

Код транс. средства	Код п. отправки	Код груза 1	Код груза 2	Код груза 3
3	2	13	19	14
4	1	8	9	12
2	2	2	3	5

Рисунок 2. Кроссинговер «обмен строками в таблицах»

Кроссинговер «обмен подстроками в таблицах» представляет собой случайный выбор подстрок, представляющих из себя несколько кодов грузов из первой хромосомы-родителя и взаимную замену с подстроками с такими характеристиками, чтобы, с учетом свободного объема и грузоподъемности транспортного средства, не нарушались соответствующие ограничения. При таком типе кроссинговера также может потребоваться корректировка хромосом после замены, однако возможна модификация этого метода, в которой выбираются будут только те грузы, которые не задействованы в перевозке в хромосоме-родителе 2.

Пример кроссинговера «обмен подстроками в таблицах» приведен на рисунке 3. Стрелкой показан обмен подстрок. При этом в данном примере предварительно было проверено ограничение на превышение суммарным объемом груза объем кузова транспортного средства и на превышение суммарным весом груза грузоподъемности транспортного средства, а также проверено ограничение на то, что один груз может быть перевезен лишь одним транспортным средством.

Хромосома-родитель 1

Код транс. средства	Код п. отправки	Код груза 1	Код груза 2	Код груза 3
1	1	13	10	4
4	2	9	7	
5	1	3	15	6

Хромосома-родитель 2

Код транс. средства	Код п. отправки	Код груза 1	Код груза 2	Код груза 3
3	2	13	19	14
4	1	16	8	12
2	2	2	3	5

Рисунок 3. Кроссинговер «обмен строками в таблицах»

Кроссинговер «комбинированный обмен строками и подстроками» является комбинацией двух предыдущих методов кроссинговера, при котором возможен одновременный обмен как строк хромосом полностью, так и подстрок, содержащих грузы, при тех же условиях, что и в двух предыдущих типах кроссинговера. При таком типе кроссинговера также может потребоваться корректировка хромосом после замены. Такой тип кроссинговера в наибольшей степени повышает вариативность хромосом – потомков, однако содержит те же недостатки, что и предыдущие два метода.

Для выхода из локального оптимума с целью глобального улучшения имеющихся решений используется мутация. Наиболее эффективным типом мутации при табличной форме хромосом является мутация «обмен подстроками», аналогичная кроссинговеру «обмен подстроками в таблицах», однако в случае мутации обмен подстроками производится между строками одной таблицы. При этом необходимо проверять выполнение тех же ограничений, что и для кроссинговера: характеристики заменяемых подстрок должны отличаться не более чем на величину меньшего свободного объема в транспортном средстве.

Пример мутации «обмен подстроками» приведен на рисунке 4. При этом в данном примере предварительно было проверено ограничение на превышение суммарным объемом груза объем кузова транспортного средства и на превышение суммарным весом груза грузоподъемности транспортного средства.

Код транс. средства	Код п. отправки	Код груза 1	Код груза 2	Код груза 3
3	2	13	19	
4	1	8	9	
2	2	2	3	5

Рисунок 4. Мутация

Редукцию с целью поддержания мощности популяции постоянной предлагается выполнять по элитарной схеме, либо по селекционной схеме. В ходе работы оператора редукции мощность популяции станет равна заданной до кроссинговера, при этом с большей вероятностью в популяции будут находиться лучшие решения из хромосом родителей, потомков и хромосом, подвергшихся мутации.

На основании полученной популяции первого уровня генетического алгоритма производится запуск генетических алгоритмов 2-го уровня, который можно выполнить параллельно при наличии достаточного количества ресурсов вычислительной системы. Входными данными алгоритмов 2-го уровня являются строки в таблицах, представляющих хромосомы алгоритма 1-го уровня. Выходными данными алгоритмов 2-го уровня являются найденные алгоритмами 2-го уровня субоптимальные стоимости путей для каждой из строк таблицы.

После завершения работы алгоритмов 2-го уровня необходимо выполнить расчет значений фитнес-функции для каждой из хромосом генетического алгоритма 1-го уровня по формуле 1.

На основании рассчитанных значений фитнес-функции необходимо проверить критерии остановки алгоритма, такие, как достижение заданного количества итераций работы алгоритма либо достижение заданного количества итераций, в течение которых

лучшее решение не изменялось. В случае если критерий останова выполнен, необходимо сохранить лучшее решение и завершить работу алгоритма.

Хромосома второго уровня генетического алгоритма более подробно рассмотрена в [4-5] и представляет собой вектор чисел, первое из которых является кодом пункта погрузки груза, остальные являются кодами пунктов выгрузки груза. Порядок следования кодов в хромосоме определяет порядок движения транспортного средства между пунктами погрузки и выгрузки. Также генетический алгоритм 2-го уровня получает информацию о времени отправки от пункта погрузки и о конечных временах выгрузки в каждом пункте выгрузки груза. На основании о временах погрузки и выгрузки выполняется проверка ограничений и отсеивание недопустимых решений.

Предлагается использование кодирования хромосом в виде «представления пути», как наиболее наглядного и простого для понимания, а также более производительного по сравнению с представлением соседства и представлением порядка [6].

Фитнесс-функция для второго этапа генетического алгоритма представляется суммарной длиной маршрута и рассчитывается по формуле 4.

$$F = \sum_{i=2}^N A_{K_{i-1}K_i} \rightarrow \min, \quad (4)$$

где F – значение фитнесс-функции, i – номер пункта выгрузки на маршруте, N – количество пунктов выгрузки на маршруте, $A_{K_{i-1}K_i}$ – расстояние между пунктами K_{i-1} и K_i из матрицы расстояний.

На втором уровне генетического алгоритма выполняется поиск решения с учетом ограничений на максимальное время доставки груза в точку разгрузки. На основании рассчитанных значений фитнесс-функции выполняется проверка критериев останова алгоритма, таких, как достижение заданного количества итераций работы алгоритма либо достижение заданного количества итераций, в течение которых лучшее решение не изменялось. В случае если критерий останова выполнен, необходимо передать в алгоритм первого уровня лучшее решение и завершить работу алгоритма 2-го уровня.

Выводы

В этой статье выполнена постановка задачи оптимизации грузоперевозок и описан разработанный двухуровневый генетический алгоритм для решения поставленной задачи. Дано описание способа кодирования решений, приведена укрупненная схема решения задачи разработанным алгоритмом. Описаны операторы разработанного алгоритма, а также приведены примеры выполнения этих операторов на наборе тестовых данных.

Литература

1. Гарнов, А. П. Инструментарий логистики: учебник / А. П. Гарнов, Н. С. Киреева – М.: Креативная экономика, 2009. – 310 с.
2. Товстик Т.М. Алгоритм приближенного решения задачи коммивояжера / Т.М. Товстик, Е.В. Жукова // Вестник СПбГУ. Серия 1. Математика. Механика. Астрономия. – СПб: СПбГУ, 2013. – с. 101-106.
3. Курейчик В.М. Генетический алгоритм решения логистической задачи / В.М. Курейчик, А.А. Рокотянский // Известия ЮФУ. Технические науки. – Таганрог: ТРТУ, 2012. – №136 – с. 245-251.
4. Александрова О.А. Оптимизация грузовых перевозок с использованием генетических алгоритмов / О.А. Александрова, А.И. Секирин // Информатика и компьютерные технологии – 2009 №5, – Донецк: ДонНТУ, с. 237–244.
5. Скобцов Ю.А. Основы эволюционных вычислений. Донецк: ДонНТУ, 2008. – 326 с.

6. Haroun S.A. A Performance Comparison of GA and ACO Applied to TSP / S.A. Haroun, B. Jamal, E.H. Hicham // International Journal of Computer Applications. – 2015. – vol.117 – pp. 28–35.

УДК 004.67

МЕТОДЫ АНАЛИЗА И ОПТИМИЗАЦИИ РАБОТЫ КУХНИ В УСЛОВИЯХ НЕПРЕРЫВНОГО ПРОИЗВОДСТВА

Озеров А.О., Хмелевой С.В.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: arkadiiozerov13@gmail.com

Аннотация:

Озеров А.О., Хмелевой С.В. Методы анализа и оптимизации работы кухни в условиях непрерывного производства. Рассмотрена предметная область и описаны причины для оптимизации. Спроектирована и разработана база данных. Проведен анализ выбранных методов и алгоритмов, которые помогут решить поставленные задачи. Произведена программная реализация методов и получены результаты их применимости.

Annotation:

Ozerov A. O., Hmelevoy S.V. Methods of the analysis and optimization of work of kitchen in the conditions of continuous production. The subject area is considered and the reasons for optimization are described. Designed and developed database. The analysis of the selected methods and algorithms that will help solve the tasks. The software implementation of the methods is made and the results of their applicability are obtained.

Общая постановка проблемы

С ростом уровня жизни и населения количество клиентов в ресторанах растет, таким образом проблема обслуживания в ресторанах стала специфическим социальным явлением. В следствии этого в последние годы рестораторы стали больше внимания уделять автоматизации своих предприятий. В условиях стабильности наибольшей популярностью пользовались ИТ-системы, которые позволяют автоматизировать самые основные функции – как правило продажи и склад. Кризис же заставил задуматься о том, как «выжать» из компании максимальную эффективность, сохранить постоянных гостей, получить необходимые оборотные средства на поддержание и развитие бизнеса [1].

Несмотря на то, что существующие информационные системы для ресторанов позволяют автоматизировать и оптимизировать многие функции, однако мало кто задумывается над оптимизацией работы кухни в следствии чего есть большая вероятность падения прибыли из-за неэффективного распределения заказов.

Следовательно, цель – минимизация времени выполнения заказа. В рамках достижения данной цели необходимо выполнение следующих задач:

- Минимизация времени выполнения заказов и их ожидания за счет эффективного распределения.
- Контроль выполнения некоторых блюд заданным множеством поваров, и контроль загрузки поваров.
- Минимизация очереди заказов.
- Минимизация времени доставки к клиенту за счет эффективного распределения между филиалами и построения оптимального пути доставки для курьера.

Исходя из данного анализа, можно сказать, что задача анализа и оптимизации работы кухни в ресторане и других заведениях общепита является актуальной на сегодняшний день.

Однако, если рассматривать данные задачи в рамках небольших заведений, она не имеет актуальности, так как из-за малых расчетов подойдут уже существующие ИС либо вообще не нужны. Поэтому задача анализа и оптимизации работы кухни будет рассматриваться в рамках крупных сетей ресторанов в густонаселенных регионах, таким примером является крупная сеть ресторанов Корчма «Тарас Бульба», которая в Москве имеет порядка 15 филиалов и способна удаленно принимать заказы.

Проектирование и разработка базы данных

Для управления БД была выбрана СУБД с реляционной моделью организации данных.

В качестве СУБД был выбран пакет MySQL

В табл. 1 представлен начальный набор сущностей, так как в процессе разработке он может меняться.

Таблица 1 – Сущности модели

№	Сущность	Описание
1	Клиенты	Информация о клиентах
2	Адрес клиента	Информация об адресе клиент
3	Районы	Справочник районов города
4	Отделы	Справочник отделов сети ресторанов
5	Повара	Информация о поварах
6	Категории	Информация о категориях продуктов
7	Продукты	Информация о продуктах
8	Заказы	Информация о заказах
9	Заказ продукты	Информация о заказанных продуктах

Физическая модель данных приведена на рис. 1.

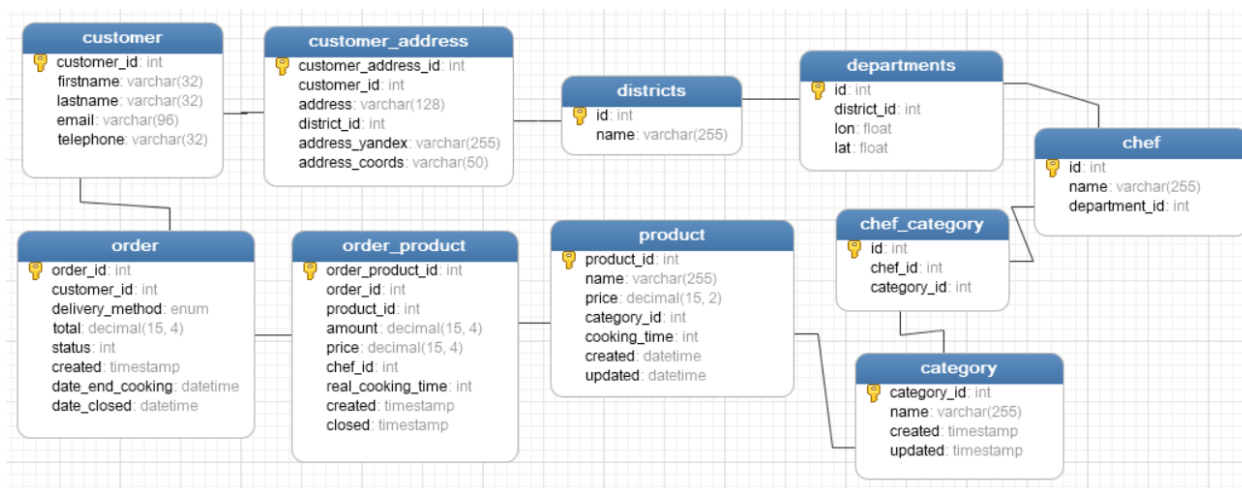


Рисунок 1 – Физическая модель данных

Обзор выбранных алгоритмов для оптимизации

Алгоритм Дейкстры – это жадный алгоритм решения задачи кратчайшего пути, который предоставляет нам кратчайший путь от одного конкретного узла источника ко всем другим узлам в данном графе.

Шаги для поиска кратчайшего пути с помощью данного алгоритма заключаются в следующем [4]:

- Нужно начать с исходного узла, обозначив его расстоянием 0, а все остальные узлы - ∞ . Исходный узел рассматривается как текущий узел и считается посещенным. Все остальные узлы считаются не посещенными.
- Выявить все не посещенные узлы, которые в настоящее время являются соседями текущего узла. Вычислить расстояние от не посещенных узлов до текущего, добавив веса их ребер.
- Пометить каждую из вершин соответствующим весом к текущему узлу, но необходимо изменить вес узла, если он меньше предыдущего значения. Каждый раз узлы маркируются своими весами
- Пометить текущий узел как посещенный. Как только вершина посещена, не нужно следить за ней снова.
- Из всех не посещенных узлов, необходимо выбрать узел, который имеет минимальный вес к текущему узлу. Далее необходимо его рассматривать как текущий рабочий узел.
- Повторять шаги 2, 3 и 4, пока не будут посещены все узлы.

Данный алгоритм имеет некоторые недостатки, такие как невозможность обрабатывать отрицательные ребра, и это приводит к ациклическим графам и чаще всего не может получить правильный кратчайший путь. Этот алгоритм сильно централизован, не очень распределен. Но его недостатки учтены в других алгоритмах, примером которых является алгоритм Флойда-Уоршелла.

На рис. 2 представлен пример начального состояния графа и конечного, после посещения всех узлов. Смотря на рис. 2, можно увидеть, что кратчайший путь от вершины 1 до 2-й составляет 7, до 3-й — 9, до 4-й — 20, до 5-й — 20, до 6-й — 11 [5].

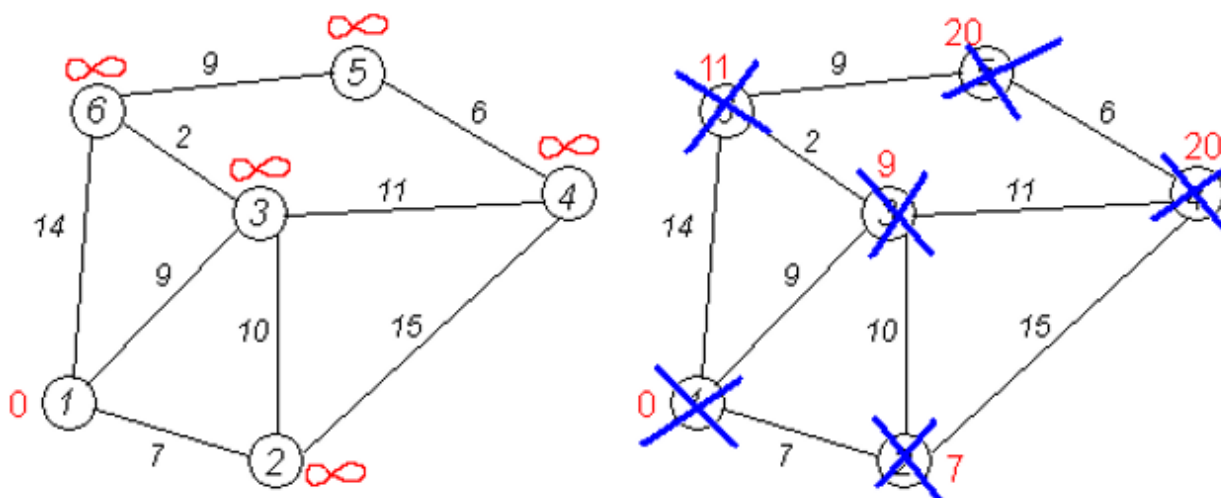


Рисунок 2 – начальное и конечное состояние графа

Алгоритм Флойда – алгоритм нахождения длин кратчайших путей между всеми парами вершин во взвешенном ориентированном графе [2].

Этот алгоритм более общий по сравнению с алгоритмом Дейкстры, так как он находит кратчайшие пути между любыми узлами сети. В этом алгоритме сеть представлена в виде квадратной матрицы с n строками и n столбцами. Элемент (i, j) равен расстоянию d_{ij} от узла i к узлу j , которое имеет конечное значение, если существует дуга (i, j) и равен бесконечности в противном случае.

Основная идея метода Флойда – есть три узла i, j и k и заданы расстояния между ними. Если выполняется неравенство $d_{ij} + d_{jk} < d_{ik}$, то целесообразно заменить путь $i \rightarrow k$

путем $i \rightarrow j \rightarrow k$. Такая замена (ее еще называют треугольным оператором) выполняется систематически в процессе выполнения алгоритма Флойда.

Алгоритм Флойда состоит из следующих шагов [3]:

- Определить начальную матрицу расстояния D_0 и матрицу последовательности узлов S_0 . Диагональные элементы матриц помечаются каким-либо знаком, который покажет, что эти элементы в вычислениях не участвуют.

- На каждом k -ом шаге задаем строку k и столбец k как ведущую строку и ведущий столбец. Рассматривается возможность применения треугольного оператора ко всем элементам d_{ij} матрицы D_{k-1} . Если выполняется неравенство $d_{ik} + d_{kj} < d_{ij}$, тогда выполняются следующие действия: создаем матрицу D_k путем замены в матрице D_{k-1} элемента d_{ij} на сумму $d_{ik} + d_{kj}$; создаем матрицу S_k путем замены в матрице S_{k-1} элемента s_{ij} на k . Полагается, что $k = k+1$ и повторяем шаг k .

- После реализации n шагов алгоритма определение по матрицам D_n и S_n кратчайшего пути между узлами i и j выполняется по следующим правилам:

- Расстояние между i и j равно элементу d_{ij} в матрице D_n .
- Промежуточные узлы пути от узла i к узлу j определяем по матрице S_n . Пусть $s_{ij} = k$, тогда имеем путь $i \rightarrow k \rightarrow j$. Если далее $s_{ik} = k$ и $s_{kj} = j$, тогда считаем, что весь путь определен, так как найдены все промежуточные узлы. В противном случае повторяем описанную процедуру для путей от узла i к узлу k и от узла k к узлу j .

Алгоритму требуется $O(n^3)$ памяти, для сохранения матриц. Однако количество матриц можно легко сократить до двух, каждый раз переписывая ненужную матрицу или вообще перейти к двумерной матрице. Лучший вариант, который чаще всего используется – писать сразу в матрицу смежности, тогда совсем не нужна дополнительная память, правда если сразу переписывать изначальную матрицу, то нужно дополнительно показать корректность алгоритма, так как классическое доказательство верно только для случая, когда матрица предыдущей итерации не изменяется. Что касается времени работы – три вложенных цикла от 1 до n – $O(n^3)$.

Пример работы алгоритма Флойда показан на рис. 3.

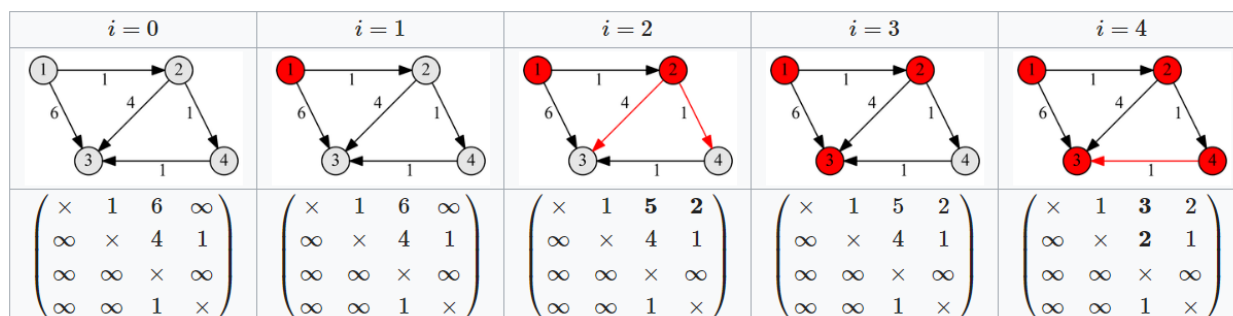


Рисунок 3 – пример работы алгоритма Флойда

Программная реализация

Описанные выше алгоритмы были реализованы на node.js. Сам процесс распределения заказов заключается в следующем:

- Определяется матрица расстояний от клиента до каждого отдела ресторана.
- Далее внутри каждого отдела вычисляется минимальное время приготовления заказа с учетом времени ожидания если все повара в данном отделе заняты.
- Так как заказ необходимо доставить до клиента, а курьер обычно развозит определенное количество заказов, то вычисляется оптимальный путь доставки заказов.

• На основании вышеперечисленных данных строятся графы, в которых в качестве веса выступает время. То есть оптимальный путь будет такой, где необходимо затратить минимальное кол-во времени.

Изначально тестовые данные показывали, что среднее время выполнения заказа (приготовление и доставка) приблизительно равнялось 1.8 часа. Во время применения алгоритма Дейкстры время сократилось до 1.6 часа. С помощью алгоритма Флойда удалось сократить время до 1.4 часа. Данные показатели и некоторые другие сведены в таб. 2.

Таблица 2 – Сущности модели

Название параметра	До оптимизации	Алгоритм Дейкстры	Алгоритм Флойда
Среднее время выполнения заказа	1.8 часа	1.6 часа	1.4 часа
Общее время выполнения тестовых заказов	1788.64 часов	1589.57 часов	1403.43 часов
Средняя очередь у поваров	8.13 чел.	6.78 чел.	6.23 чел.

Выводы

В результате проведенных исследований и анализа показателей тестовых данных до оптимизации и после с помощью алгоритмов Дейкстры и Флойда была доказана целесообразность применения данных алгоритмов к распределению заказов. Так же было выяснено, что метод Флойда по сравнению с Дейкстры лучше определяет оптимальный путь. Однако алгоритм Флойда так же имеет свои недостатки. Поэтому в дальнейшем для оптимизации распределения заказов будет взят за основу достаточно новый метод «The Rectangular algorithm». Он требует меньше вычислительных усилий из-за меньшего объема вычислений. Кроме того, он гораздо легче в понимании [7].

Литература

1. Дизайн ресторана системы массового обслуживания: [Электронный ресурс]. URL: <https://swsu.ru/sbornik-statey/dizayn-restorana-sistemy-massovogo-obsluzhivaniya.php>.
2. Алгоритм Флойда: [Электронный ресурс]. URL: <http://neerc.ifmo.ru/wiki/index.php?title=%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%A4%D0%BB%D0%BE%D0%B9%D0%B4%D0%B0>.
3. Принцип работы алгоритма Флойда: [Электронный ресурс]. URL: http://khpi-iip.mipk.kharkiv.edu/library/datastr/book_sod/kgsu/din_0124.html.
4. Using Dijkstra's Algorithm: [Электронный ресурс]. URL: <http://www.iosrjournals.org/iosr-jce/papers/Vol20-issue2/Version-2/B2002021015.pdf>.
5. Алгоритм Дейкстры: [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%94%D0%B5%D0%B9%D0%BA%D1%81%D1%82%D1%80%D1%8B>
6. Алгоритм Флойда-Уоршелла: [Электронный ресурс]. URL: <https://habr.com/ru/post/105825/>
7. Speeding up the Floyd-Warshall algorithm for the cycled shortest path problem: [Электронный ресурс]. URL: <https://www.sciencedirect.com/science/article/pii/S0893965911002928>

УДК 04.943

ОПРЕДЕЛЕНИЯ КОЛИЧЕСТВА ТРАНСПОРТНЫХ СРЕДСТВ С ПОМОЩЬЮ ИМИТАЦИОННОЙ МОДЕЛИ И МЕТОДОВ ПРИНЯТИЯ РЕШЕНИЙ

Шиленко А.В., Светличная В.А.

Донецкий национальный технический университет
кафедра автоматизированных систем управления
E-mail: sashazemzer@gmail.com

Аннотация:

Шиленко А.В., Светличная В.А. Определения количества транспортных средств с помощью имитационной модели и методов принятия решений. В данной статье представлена задача определения количества элементов транспортной системы с использованием методов имитационного моделирования и методов теории принятия решения. Установлены основные альтернативы и критерии. В результате найдены наиболее оптимальные значения искомых параметров объекта, удовлетворяющие выбранным критериям.

Ключевые слова: имитационное моделирование, формализация, метод принятия решения, альтернатива, критерий.

Annotation:

Shilenko A.V., Svetlichnaya V.A. Quantification of transport means using the simulation model and methods of decision-making. The article presents the task of identifying the number of transportation system components using the simulation modelling techniques and methods of decision theory. The main alternatives and criteria are identified. As a result, the most optimal values of required object attributes satisfying criteria selected are found.

Key words: simulation modelling, formalization, methods of decision-making, alternatives, criterion.

Общая постановка проблемы

Принятие решений – основа любого управления. Обоснование и выбор конкретных управленческих решений, связанных с финансовыми рисками, базируется на концепции и методологии теории принятия решений [1].

Теория принятия решений представляет собой совокупность математических и численных методов, ориентированных на нахождение наилучших вариантов из множества альтернатив и позволяющих избежать их полного перебора.

В теории принятия решений используются процедуры выбора наилучшей из нескольких возможных альтернатив. С этой точки зрения процесс принятия решений может принадлежать к одному из трех возможных условий: - принятие решений в условиях определенности, когда данные известны точно; - принятие решений в условиях риска, когда данные можно описать с помощью вероятностных распределений; - принятие решений в условиях неопределенности, когда данным нельзя приписать относительные веса, которые представляли бы степень их значимости в процессе принятия решений [2].

Насколько правильным будет выбор, зависит от качества данных, используемых при описании ситуации, в которой принимается решение. На практике не всегда существует возможность получить достаточное количество достоверных данных, характеризующих исследуемый объект. Одним из вариантов получения качественных данных для

использования их в методах теории принятия решений является математическое имитационное моделирование. Использование имитационного моделирования для статистического отображения функционирования объекта и получения статистических данных для выбора и реализации методов принятия решений позволяет на этапе проектирования или реорганизации объекта позволяет решить ряд задач по структуре или организации работы объекта. Преимуществом использования модели является получение информации в любых условиях функционирования рассматриваемого объекта [3,4]. При таком подходе решение задачи состоит из двух частей.

В первую очередь разрабатывается имитационная модель, с помощью которой накапливается требуемая качественная статистическая информация, характеризующая функционирование объекта в различных условиях, выделенных как альтернативы.

На следующем шаге с помощью методов теории принятия решения производится выбор оптимальной альтернативы.

Исследования

Постановка задачи.

В качестве объекта анализа рассматривается транспортное предприятие, автомобили которого развозят продукцию потребителям. Предприятие имеет определенное количество погрузочных устройств kz . Изделия развозятся кр потребителям, причем способ доставки может быть различным: - только одному потребителю один тип изделия; - один тип изделия несколькими потребителям; - несколько типов изделия одному потребителю. Требуется найти оптимальное количество грузовых автомобилей в данной системе для перевозки изделий потребителям таким образом, чтобы при наименьшем их количестве и предприятие, и потребители не испытывали недостатка в реализации или потреблении продукции.

Решение задачи.

Согласно поставленной задаче, решение задачи содержит два этапа. На первом этапе разрабатывается моделирующий алгоритм, с помощью которого собираются статистические данные, характеризующие функционирование объекта в различных условиях. Условия работы исследуемого объекта выделены как альтернативы. Полученные путем статистического имитационного моделирования данные служат основой для выбора оптимальной альтернативы, которая определяется на втором этапе путем использования методов теории принятия решения.

Анализ функционирования рассматриваемого объекта, позволил отнести его к классу замкнутых систем массового обслуживания. При формализации предприятие и потребители рассматривались как фазы замкнутой системы массового обслуживания. В качестве транзактов (заявок) служат автомобили. Для учета возможных поломок, последние также формализовались как заявки с законом поступления, отличающимся от закона поступления автомобилей. С использованием вероятностных законов и характеристик объекта, был разработан моделирующий алгоритм, представленный на рисунке 1.

В программе рассчитываются такие статистические данные как:

- среднее время ожидания машин производителем;
- среднее время ожидания машин потребителями (в задании звучит как «интервал доставки товара»);
- среднее время простоя машин перед загрузкой у производителя;
- среднее время ожидания разгрузки машин у потребителя.

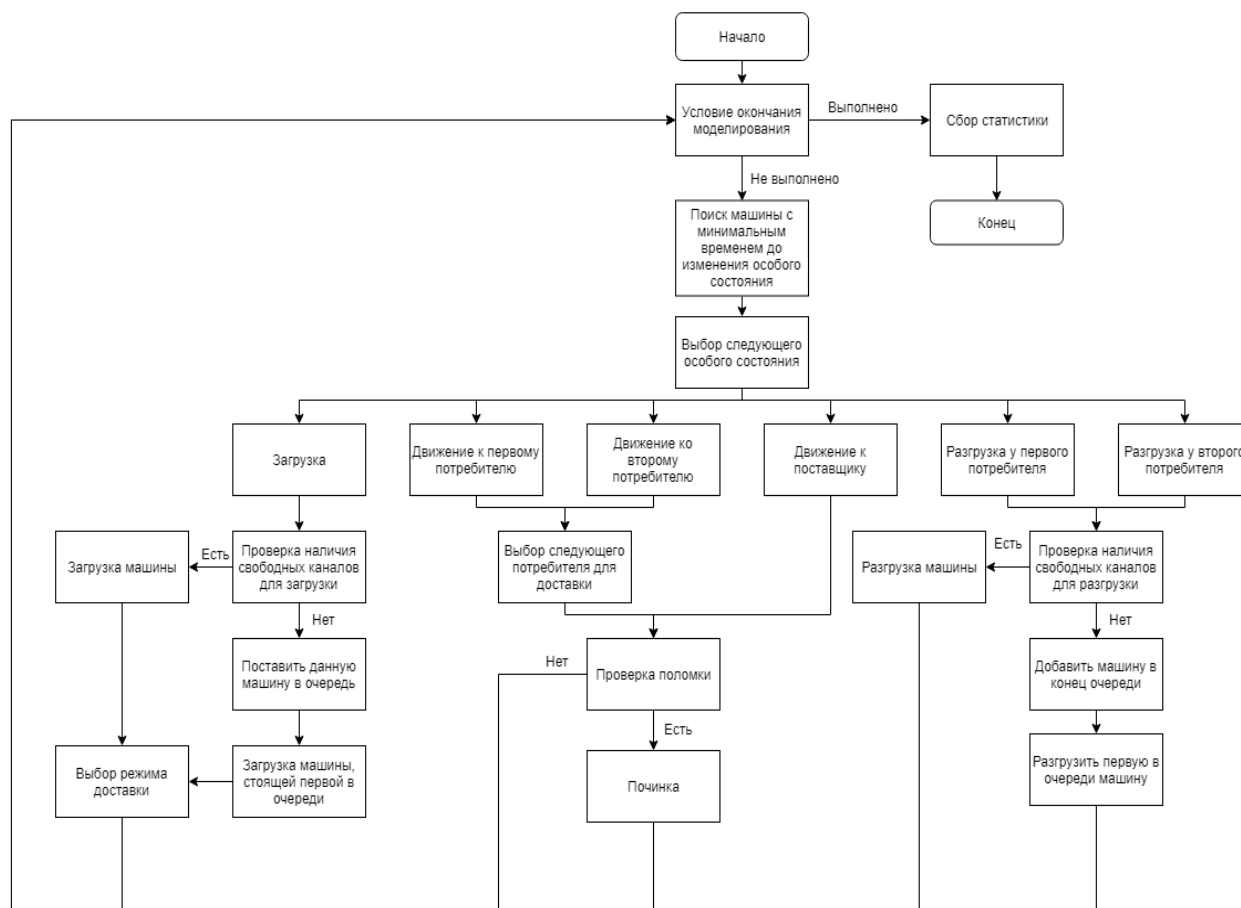


Рис. 1. Укрупненная схема моделирующего алгоритма

Для поиска оптимального решения была построена таблица 2, содержащая в себе критерии (C_i) и альтернативы (A_i). В качестве альтернатив принималось различное количество грузовиков, в качестве критериев перечисленные статистические параметры.

Необходимо выбрать ту альтернативу A_i , которая даст наименьшее соотношение простоя, ожидания и количества задействованных машин. Для этого использовались методы решения многокритериальных задач. Результаты, полученные с помощью имитационного алгоритма, приведены в таблице 1.

Таблица 1 – Данные, полученные с помощью имитационного алгоритма

	C1	C2	C3	C4	C5
A1	5,74	1,12	8,52	83,87	6
A2	7,44	1,2	2,97	67,86	7
A3	9,53	2,36	6,18	65,77	8
A4	19,22	3,43	3,67	58,18	9
A5	18,58	3,49	1,82	47,99	10
A6	24,15	4,4	5,8	63,97	11
A7	29,15	4,48	4,16	56,9	12

Для нахождения оптимальной альтернативы были использованы следующие методы: адаптивной сверки, мультипликативной сверки, главного критерия, уступок, целевого программирования, гарантированного результата.

В качестве примера решения задачи поиска оптимальной альтернативы приведем использование метода аддитивной сверки. Для начала необходимо нормализовать критерии. Делается это для того, чтобы была возможность сравнивать несравнимые величины, имеющие различные единицы измерения. В данном методе нормализация критериев, стремящихся к минимуму, выполняется по формуле 1.

$$c'_i(X) = \frac{c_i^{\max} - c_i(X)}{c_i^{\max} - c_i^{\min}} \quad (1)$$

Для нормализации необходимы дополнительные данные, а именно минимальное, максимальное значения в столбце и разница между ними (таблица 2).

Таблица 2 – Граничные значения столбцов и их разница

	C1(min)	C2(min)	C3(min)	C4(min)	C5(min)
min	5,74	1,12	1,82	47,99	6
max	29,15	4,48	8,52	83,87	12
Разница	23,41	3,36	6,7	35,88	6

Далее необходимо расставить приоритеты (α) для каждого критерия. Сумма значений α не должно превышать 1. Поскольку основной задачей данной статьи является определение минимального количества машин, позволяющее функционировать системе максимально синхронно, для критерия C5 был выбран приоритет 0.4. Так же было принято решение как можно больше уменьшить ожидание загрузки и разгрузки машин, поэтому для критериев C1 и C2 выбран приоритет 0.2. C3 и C4 получили по 0.1 приоритета. Для поиска оптимального решения необходимо добавить столбец в уже нормализованную матрицу и записать туда значения, рассчитанные по формуле 2.

$$f(x) = \sum a_k f_k \quad (2)$$

Наибольшее значение в новом столбце будет определять оптимальную альтернативу. Результат данных действий показан в таблице 3.

Таблица 3 – нормализация и вычисление дополнительного столбца

a	0,2	0,2	0,1	0,1	0,4	
	C1(min)	C2(min)	C3(min)	C4(min)	C5(min)	AC
A1	1	1	0	0	1	0,8
A2	0,927381	0,97619	0,828358	0,44621	0,8333333	0,8415045
A3	0,838103	0,630952	0,349254	0,504459	0,6666667	0,645849122
A4	0,424178	0,3125	0,723881	0,715998	0,5	0,491323377
A5	0,451516	0,294643	1	1	0,3333333	0,482565194
A6	0,213584	0,02381	0,40597	0,554627	0,1666667	0,210205027

A7	0	0	0,650746	0,751672	0	0,140241851
Решение						0,8415045

Аналогичным образом были проведены исследования и выполнены вычисления с помощью указанных методов теории принятия решений. Поиск оптимальной альтернативы шестью методами показал схожий результат. Из них 5 методов (аддитивной свертки, гарантированного результата, последовательных уступок, главного критерия, целевого программирования) указали на альтернативу A2, которая соответствует 7 машинам и 1 метод (мультипликативной свертки) указал на альтернативу A1, соответствующую 6 машинам. Однако, при должном округлении результатов в мультипликативной свертке (хотя бы до 2х знаков), результаты альтернатив A1 и A2 равны и выбор остается за лицом, принимающим решение (ЛПР). В результате был сделан вывод о том, что искомой альтернативой является альтернатива A2.

Выводы

В результате выполнения исследований был разработан моделирующий алгоритм системы массового обслуживания замкнутого типа. На базе статистических данных, собранных путем его реализации, была построена таблица критериев и альтернатив, с помощью которой и определено субоптимальное решение задачи минимизации количества используемых грузовых машин. Результаты исследований могут быть использованы как для определения количества транспортных средств, так и для организации работы поставщиков и потребителей.

Литература

1. Принятие решений в условиях риска и неопределенности решений / Интернет-ресурс. - Режим доступа: <http://www.management.com.ua/qm/qm215.html> . - Загл. с экрана.
2. Center YF/ Принятие решений / Интернет-ресурс. - Режим доступа: <https://center-yf.ru/data/ip/prinyatie-resheniy.php> . - Загл. с экрана.
3. Андриевская А.В. Использование имитационного моделирования в компьютеризированной системе управления транспортными потоками в условиях большого города / А.В. Андриевская, В.А. Светличная, Н.К. Андриевская // Мир компьютерных технологий. Сборник статей всероссийской студенческой научно-технической конференции. Научный редактор Е.Н. Машенко. - 2018, - С. 7-12.
4. Левченко Д.И. Разработка имитационной модели для анализа динамики движения материалов и сырья с учетом случайных факторов / Д.И. Левченко, В.А. Светличная, К.К. Бабич // Информатика и кибернетика. -Д.: ДонНТУ, 2016. - № 3 (5). - С. 57-64.
5. Ковальчук Н.А. Подход к проектированию информационной системы поддержки принятия решений в задачах социальной служб / Н.А. Ковальчук, О.В. Ченгарь // Мир компьютерных технологий. Сборник статей всероссийской студенческой научно-технической конференции. Научный редактор Е.Н. Машенко - 2018, - С. 64-67.
6. Полковниченко Я.Д., Светличная В.А., Андриевская Н.К.. Применение имитационной модели для принятия решения по структуре оборудования рудного карьера // Сборник Всероссийской научно-технической конференции студентов, аспирантов и молодых ученых « В мире компьютерных технологий», г.Севастополь, 2-5 апреля 2019г.

УДК 004.94

ИСПОЛЬЗОВАНИЕ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ПРИ КАЛЕНДАРНОМ ПЛАНИРОВАНИИ

Стрельников Е.А., Светличная В.А.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: shunterisuus@gmail.com

Аннотация:

Стрельников Е.А., Светличная В.А. Использование имитационного моделирования при календарном планировании. В статье рассматривается формализация технологического процесса обработки деталей в виде многофазной системы массового обслуживания с целью разработки производственной программы механообрабатывающего цеха.

Annotation:

Strelnikov E.A. Svetlichnaya V.A. The use of simulation in scheduling. The article discusses the formalization of the technological process of processing parts in the form of a multi-phase queuing system in order to develop the production program of the machining shop.

Общая постановка проблемы

В условиях стремительного развития технологических процессов в машиностроении становится актуальной автоматизация и совершенствование методов оперативно-календарного планирования, которое обеспечивает создание производственных заданий и организацию их выполнения. Оперативное управление производством выполняется благодаря реализации таких функций управления как оперативное планирование, контроль и диспетчерское регулирование производства. Главными из них являются функции оперативного планирования и диспетчерского регулирования, в процессе реализации которых принимаются решения по управлению производством.

Своевременность и качество обработки заказов, эффективность использования рабочей силы и основных фондов обуславливается точностью и качеством оперативно-календарных планов.

Отдельно следует выделить заказы предприятий, направленные на ремонт различных элементов или частей крупных производственных комплексов, например, заказы, связанные с ремонтом двигателей различного типа и мощности. Выполнение таких работ составляют мелкосерийные и единичные заказы, календарное планирование для которых является более сложным и трудоемким, чем в случае серийного производства, поскольку организация выполнения каждого нового заказа, отличается от предыдущих.

В связи с этим приходится часто сталкиваться с задачами календарного планирования наиболее рациональной загрузки производственных мощностей в разное время и решать их.

Исследования

Постановка задачи

Одним из основных процессов, составляющих заказы по ремонту оборудования, является процесс механической обработки деталей. Такой технологический процесс является достаточно сложным объектом управления, ощущающий влияние большого количество различных факторов. Вследствие этого, появляется необходимость часто и в короткие

промежутки времени решать специфические задачи планирования оптимальной загрузки механообрабатывающего оборудования, что усложняет применение известных методов и моделей.

Главными задачами календарного планирования (КП) производства являются: – распределение объема работ каждой позиции плана; – определение очередности выполнения работ; – определение сроков выполнения работ. Наиболее используемыми в настоящее время методами КП производства являются: - **ленточные графики Ганта, - цикловые графики, - объемно-календарные графики, - метод сетевого планирования производства, -методы моделирования [1].** Недостатками таких методов является **трудоемкость при необходимости внесения изменений значений каких-либо параметров процесса обработки. Использование же имитационного моделирования позволяет устранить этот недостаток, а также собрать необходимую статистическую информацию о загрузке оборудования, выявлении узких мест и вариантов их устранения или сокращения.**

Решение задачи

Для решения задачи оперативно-календарного планирования цеха машиностроительного предприятия было использовано имитационное моделирование.

После анализа процесса функционирования механизированного цеха машиностроительного предприятия он был формализован как многофазная система массового обслуживания. В качестве транзактов (заявок) принимаются партии деталей, поступающие на обработку. Многофазные СМО состоят из нескольких типовых узлов (фаз), расположенных последовательно. Каждая фаза содержит каналы (один или несколько), в качестве которых в данной модели выступает непосредственно технологическое оборудование, а именно станки. Все заявки, обслуженные в одном узле, направляются в следующий узел. Другими словами, выходной поток одного узла многофазной СМО является входным потоком для следующего. Однако порядок прохождения узлов может быть различным для разных заявок. Для части заявок может требоваться обслуживание во всех узлах, а для других – только в некоторых из узлов. Узлами в данной работе являются группы станков, на которых может быть выполнена та или иная операция для партии деталей. Количество узлов и последовательность зависит от типа детали и её технологического маршрута обработки. Особенностью рассматриваемого технологического процесса является наличие транспортных тележек, которые перевозят детали от одной фазы обработки к другой. Их функционирование также формализовано как фазы.

Структуру моделируемого процесса можно изобразить в виде потока, проходящего через различные элементы СМО (см. рис.1)

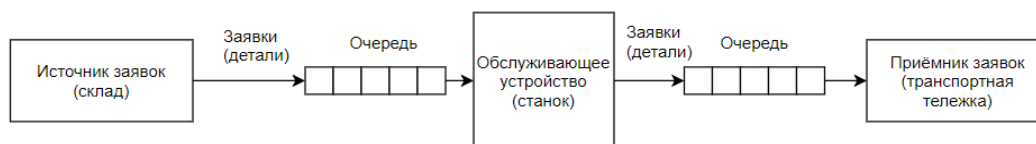


Рисунок 1 – Структура моделируемого процесса

Деятельность механического цеха начинается с того, что происходит формирование начальных партий деталей для обработки. В каждой партии содержатся однотипные детали, технологический маршрут обработки которых задается в виде цепочки из видов оборудования, закрепленного за выполнением определенных типов операций технологического процесса. После формирования производственной программы выполняется загрузка партий деталей во входные накопители станков. Станок, являясь обслуживающим устройством в системе, выполняет обработку деталей.

Партии деталей принимаются к обработке в порядке поступления. В системе происходят такие события, как поступление заявок в накопитель, постановка партий в очередь, начало обработки, обработка, конец обработки, помещение в выходной накопитель, транспортирование в следующий узел или на склад и т.д.

Укрупненная формализованная блок-схема моделирования технологического процесса в цеху машиностроительного предприятия выглядит следующим образом (см. рис.2).



Рисунок 2 – Укрупненная блок-схема моделирующего алгоритма технологического процесса

В начале технологического процесса в цеху машиностроительного предприятия нужно сформировать партии деталей для запуска таким образом, чтобы максимально задействовать имеющееся оборудование. Размер партии определяется размером входного накопителя, располагающегося возле станка. Во время выполнения операции станок имеет состояние «занят», в любое другое время состояние «свободен». Если обслуживающее устройство свободно, то оно принимает партию деталей к обслуживанию. После обработки заявок на текущем узле производится транспортировка деталей к следующему узлу либо на склад, если были выполнены все необходимые операции над деталями.

В качестве метода моделирования применен метод особых состояний, который заключается в последовательном рассмотрении особого состояния с наименьшим временем наступления. В качестве особых состояний для разрабатываемого моделирующего алгоритма выбраны моменты окончания обработки партии деталей на станках, которые, в свою очередь, являются моментами запроса на перевозку в следующую фазу, то есть моментами запроса начала работы транспортных тележек. С другой стороны, момент освобождения станка от обработки партии деталей является запросом на обработку следующей партии деталей, согласно производственной программе. Таким образом, общий моделирующий

алгоритм должен последовательно, в зависимости от момента наступления, запускать модуль моделирования работы станка той или другой фазы, или модуль моделирования работы транспортной тележки, обслуживающей текущую и последующую фазы, или склад.

Блок-схема модуля моделирования работы станка в общем виде представлена на рисунке 3, где t_n – время начала обработки партии деталей, t_d – время освобождения партии деталей, t_m – время освобождения станка.

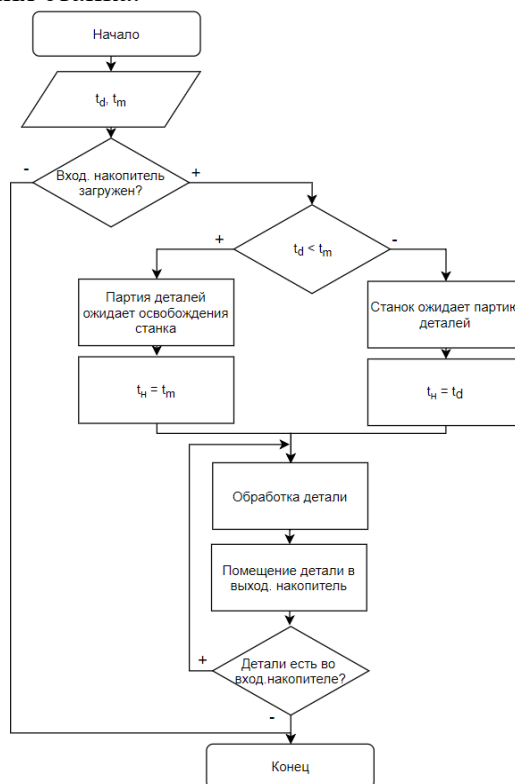


Рисунок 3 – Модуль моделирования работы станка

Аналогичным образом выглядит модуль моделирования работы тележки.

Использование приведенного моделирующего алгоритма позволяет не только планомерно распределить выполнение производственной программы по станкам и по времени, но и позволит получить некоторые интегральные характеристики работы станков и цеха в целом. Одним из наиболее важных таких характеристик является коэффициент загрузки станков. Кроме этого существует возможность анализировать работу транспортных тележек, организовывать рациональный подвоз деталей из склада и на склад.

Данный алгоритм является частью программного обеспечения компьютерной системы управления загрузкой оборудования механообрабатывающего цеха машиностроительного предприятия [3]. Для его реализации разработана база данных, хранящая в себе данные, необходимые для проведения моделирования:

- Справочник деталей (tDetails);
- Справочник станков (tMachines);
- Справочник операций (tOperations);
- Справочник функционала станков (tMachineOperations);
- Технологические маршруты для деталей (tTechRoute);
- Данные о заказах (tOrders);
- Данные о складе (tWarehouse);
- Данные о моделировании (tModelir, tModelirDetails, tModelirMachines, tModelirCart).

Алгоритмы работы всех модулей и сам процесс моделирования реализованы на языке C#, в качестве СУБД использовался пакет MySQL. Все модули реализованы с использованием объектно-ориентированного подхода.

Результаты тестирования моделирующего алгоритма представлены на рисунках 4-5.

Время	Событие
00:38	5 Токарный 16T02П обработал деталь 77
00:38	33 Фрезерный с ЧПУ F80 обработал деталь 177
00:39	33 Фрезерный с ЧПУ F80 обработал деталь 178
00:39	13 Токарный с ЧПУ C16A16Ф3 обработал деталь 83
00:39	35 Фрезерный с ЧПУ F80 обработал деталь 181
00:39	15 Токарный с ЧПУ C16A16Ф3 обработал деталь 86
00:39	13 Токарный с ЧПУ C16A16Ф3 обработал деталь 84
00:40	Деталь загружена во входнакопитель у 37 Фрезерный с Ч...
00:40	15 Токарный с ЧПУ C16A16Ф3 обработал деталь 87
00:40	Деталь загружена во входнакопитель у 37 Фрезерный с Ч...
00:40	Деталь загружена во входнакопитель у 37 Фрезерный с Ч...
00:40	Деталь загружена во входнакопитель у 37 Фрезерный с Ч...

Рисунок 4 – Результат хода моделирования работы станков

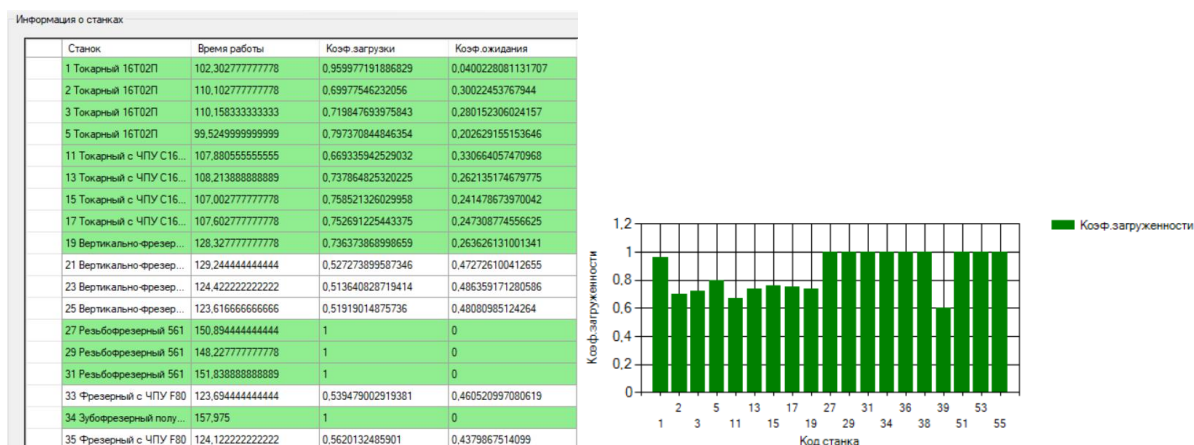


Рисунок 5 – Результаты оценки загрузки станков

Выводы

Разработанная имитационная модель позволяет промоделировать в реальном времени либо проиграть с определённого момента технологический процесс на разных входных данных, составить оперативно-календарный план на заданный период времени или для обработки заданной производственной программы, а также оценить работу технологической цепочки.

Литература

7. Методы календарного планирования производства / Интернет-ресурс. - Режим доступа: <http://www.economic-s.ru/index.php/theory/operativnoe-planirovanie-opp/metodyi-kalendarnogo-planirovaniya-pr/>. - Загл. с экрана.
8. Тюленев Л.В. Организация и планирование машиностроительного производства: Учебное пособие / Л.В. Тюленев. – СПб: Бизнес-пресса, 2001. – 304 с.
9. Стрельников Е.А., Светличная В.А., Шевченко Д.Д. Анализ методов сетевого планирования для АСУ нагрузкой механического оборудования // Материалы IX Международной научно-технической конференции «Информатика, управляющие системы, математическое и компьютерное моделирование» (ИУСМКМ-2018). – Донецк: ДонНТУ, 2018. – с. 11-15

УДК 004.4'22

РАЗРАБОТКА ФУНКЦИОНАЛЬНОЙ МОДЕЛИ WEB-ОРИЕНТИРОВАННОЙ СИСТЕМЫ БРОНИРОВАНИЯ РАЗЛИЧНЫХ ТУРИСТИЧЕСКИХ УСЛУГ

Ясницкий М.В., Андриевская Н.К.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: max.yasnitsky@gmail.com

Аннотация:

Ясницкий М.В., Андриевская Н.К. Разработка функциональной модели Web-ориентированной системы бронирования различных туристических услуг. Описана технология бронирования билетов и мест в гостинице с помощью GDS и ADS. Проведен сравнительный анализ системы с существующими аналогами. Разработана функциональная модель Web-системы. Выбраны средства для реализации, а также разработана структура программных модулей, соответствующая разработанной функциональной модели.

Annotation:

Yasnitsky M.V. Andrievskaya N.K. Development of a functional model of a Web-based reservation system for various tourist services. The technology of booking tickets and hotel rooms using GDS and ADS is described. A comparative analysis of the system with existing analogues. Developed a functional model of the Web-system. Selected tools for implementation, have also developed a functional model.

Общая постановка проблемы

В наше время многие, отправляясь в путешествие, отдают предпочтение бронированию жилья и билетов через интернет. Первые компьютерные системы резервирования CRS (Computer Reservation System), появившиеся в середине 60-х гг. XX в. и позволили ускорить процесс бронирования авиабилетов и осуществить его в режиме реального времени. Вскоре компьютерные системы бронирования в рамках одной авиакомпании стали объединяться в глобальную систему бронирования различных туристских услуг (Global Distribution Systems). На российском рынке представлены в основном такие системы глобального бронирования, как Amadeus, Galileo, Worldspan. Компьютерная система бронирования Amadeus функционирует на российском рынке с 1993 г., и на данный момент в России насчитывается более 600 турфирм — пользователей системы[1].

В противовес GDS, доступ к которым имеют лишь агенты туристических компаний, ADS (Альтернативные Системы Бронирования) предоставляют услуги по бронированию услуг частным клиентам. Заходя на любой из порталов ADS, клиент может самостоятельно выбрать нужную ему гостиницу и тип номера на определенные даты, забронировать размещение в режиме реального времени и получить моментальное подтверждение о бронировании на свой электронный ящик[2].

Существуют еще и специальные туристические сервисы, которым доступны услуги GDS и ADS. Туристский Интернет-портал — крупный сайт, предлагающий посетителю разнообразные услуги, в том числе и платные (поиск туров, заказ билетов, поиск отеля и др.) и содержащий разнообразную информацию по туристской тематике (описания стран, перечень турфирм, отзывы туристов, информация о погоде, курсы валют, различные справочники и др.). Технология заказа/покупки билетов с использованием таких сервисов приведена на рисунке 1. Аналогично выполняется и бронирование жилья, а также других

дополнительных услуг. В таком случае отсутствует «привязанность» к турфирме и присутствует полная свобода выбора, а также экономия средств на туристической услуге.

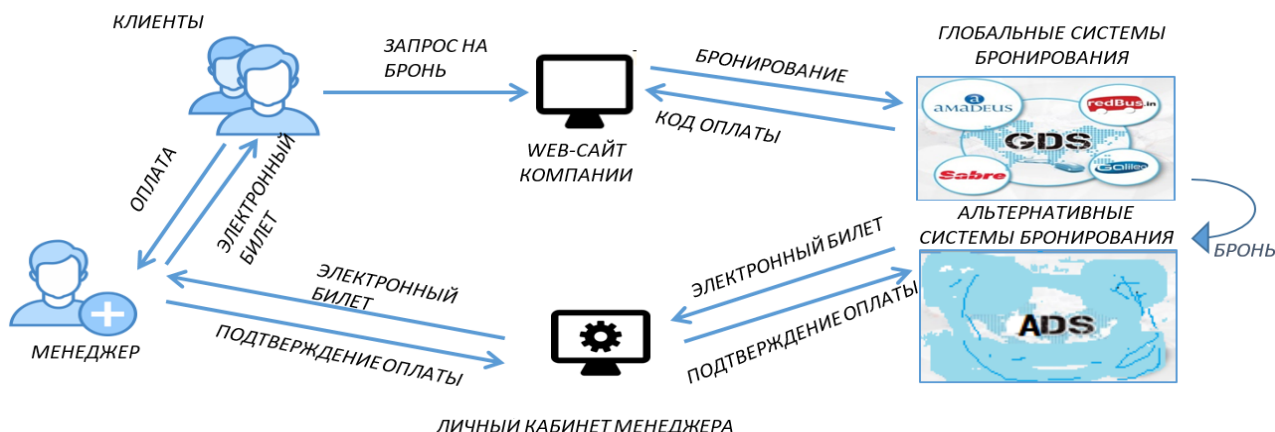


Рис. 1. Технология бронирования билетов

В следствие этого возникает необходимость разработки Web-ориентированной системы, предоставляющей возможности обширного поиска и бронирования жилья, а также предоставления выбора трансфертных и других дополнительных услуг.

Исследования

Целью исследований является разработка функциональной структуры WEB-системы, позволяющей объединить и упростить бронирование жилищных и других дополнительных услуг.

При изучении аналогов разрабатываемой системы, были выделены следующие 3 системы - Airbnb, Ostrovok.ru, Hotels.com. В таблице 1 рассмотрены возможности систем аналогов и таким образом доказана необходимость разработки новой системы.

Таблица 1 – Сравнение функционала аналогов системы

Функции\Система	Airbnb	Ostrovok.ru	Hotels.com	Система
Бронирование номера	+	+	+	+
Учет ранее использованных услуг клиентом	-	-	-	+
Учет клиентов	+	+	+	+
Учет гостиниц	+		+	+
Покупка авиабилетов	-	+	-	+
Системы оплаты	Apple Pay, Google Pay, PayPal, карты	PayPal, карты	Карты	Карты, Web Money, PayPal, Yandex Money
Учет геолокации	-	-	-	+
Заказ такси от/до аэропорта	-	-	-	+
Сдача в аренду жилищного фонда	-	-	+	+

Таким образом, WEB-ориентированная система будет представлять собой интернет площадку с возможностью поиска и выбора в первую очередь временного жилья.

Пользователь устанавливает критерии для поиска и выбора жилья и дополнительных жилищных услуг в заданных им самим странах, городах, заказывает авиабилеты для осуществления их бронирования или покупки, оставляет заявку на предоставление транспортных услуг такси (или же другого транспорта, что будет добавлено в будущем), а также пользователь может выставить свою жилищную площадь для аренды. Каждый пользователь обязан будет иметь персональный аккаунт для использования и управления предоставляемых сайтом услуг.

Для описания функциональности ИС и взаимодействия с пользователями и внешним миром на практике используются диаграммы вариантов использования USE-CASE[3-4], однако возможно разработку функциональной модели проводить по стандарту IDEF0 с помощью CASE-средства BPWin. С целью отображения взаимодействия подсистемы с внешним миром построена контекстная диаграмма, которая является вершиной древовидной структуры диаграмм стандарта IDEF0 (рис. 2).

На вход поступает: информация о клиентах, информация об аэропортах, информация о гостиницах, информация о такси, данные о заказе, данные об оплате.

На выходе системы формируются заявки на бронирование номера, заявки на заказ такси или бронирование/покупку авиабилета. Заявки на бронирование/покупку отсылаются системой автоматически. Составляется накладная на оплату. Формируется список клиентов и разнообразные отчеты.

В качестве управляющих воздействий на систему можно выделить действующее законодательство и отраслевые нормативные документы. Работают с системой клиенты и менеджеры.

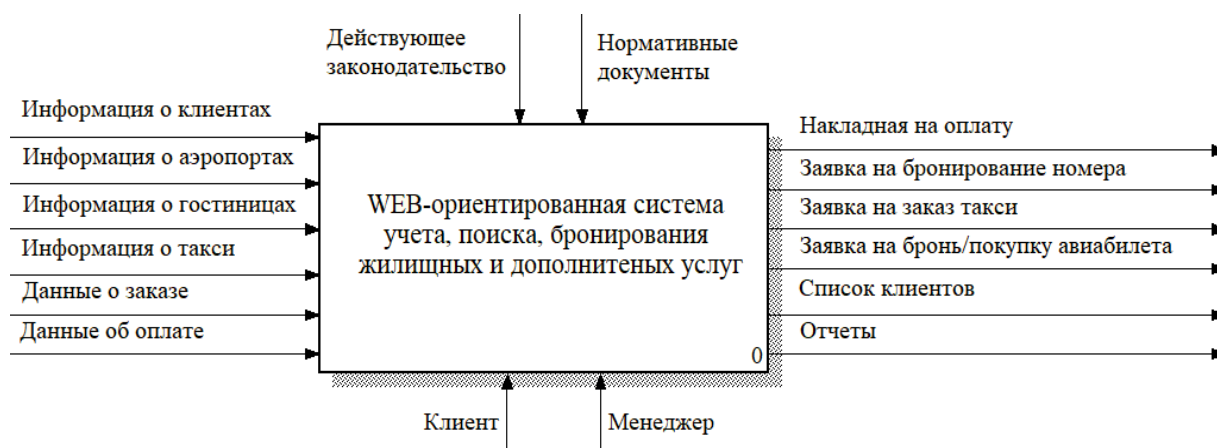


Рис. 2. Контекстная диаграмма по стандарту IDEF0

При декомпозиции первого уровня были выделены следующие функции (см. рис. 3):

- Учет клиентов. В данной функции происходит регистрация клиента и формирование списка клиентов.

- Учет гостиниц. В данной функции происходит учет всех возможных гостиниц. Она необходима для формирования списка гостиниц, для отображения их клиенту.

- Учет заказов. Функция необходимая для формирования учета использованных услуг пользователем. В ней происходит формирование данных о заказах и данные о использованных услугах.

- Расширенный поиск. В данной функции производится поиск по трансфертным и жилищным услугам, которыми можно будет воспользоваться по отдельности.

- Формирование заказа. В данной функции происходит формирование заказа по указанным параметрам пользователя. После формирования заказа данные о заказах

поступают в функцию «учет заказов». Также выходят заявки на бронирование услуг, выбранных пользователем и накладная на оплату выбранных услуг.

- Подтверждение оплат. Происходит подтверждение заказа оплатой пользователя, после чего отправляются заявки на бронирование или оплату предоставленных услуг.

- Формирование отчетности. Функция в которой происходит формирование отчетности о работе системы. Менеджер обрабатывает поступившие в нее данные.

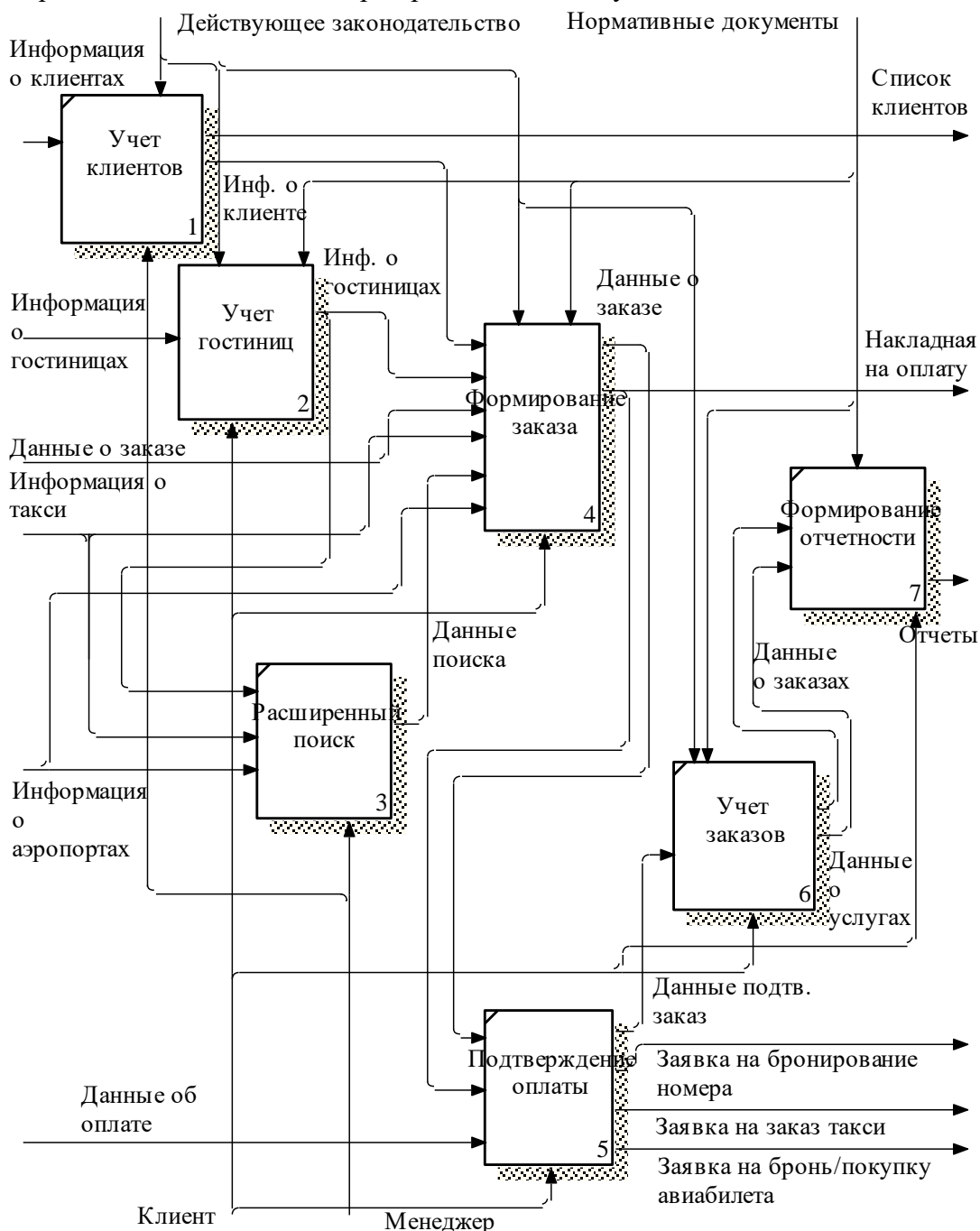


Рис. 3. Декомпозиция первого уровня по стандарту IDEF0

Для реализации данной WEB-системы будет использован язык программирования PHP версии 7.2, 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].

Для функционирования базы данных WEB-системы будет использоваться свободная реляционная система управления базами данных СУБД MySQL версии 5.7.20 [6].

Для реализации интерфейса системы будет использоваться HTML и CSS.

HTML - стандартизированный язык разметки документов во Всемирной паутине. Большинство веб-страниц содержат описание разметки на языке HTML (или XHTML) [7]. CSS - формальный язык описания внешнего вида документа, написанного с использованием языка разметки [8].

Для упрощения реализации некоторых частей системы будут использоваться готовые решения в виде фреймворка Laravel. Laravel - бесплатный веб-фреймворк с открытым кодом, предназначенный для разработки с использованием архитектурной модели MVC [9].

Разрабатываемая WEB-система будет состоять из следующих модулей:

1) *Проживание*. В данном модуле пользователь может выбрать место и длительность проживания, а также предоставленные дополнительные услуги жилья. Для отображения жилья пользователю будет использоваться функция «учет гостиниц». При выборе пользователем жилья отсылается заявка на бронирование, затем пользователю приходит уведомление о состоянии бронирования.

Что касается модуля бронирования, то принципиально новый фундамент создавать не нужно, есть уже проверенные решения, например, в виде Channel manager.

Channel manager – это программное обеспечение для ведения учета работы гостиниц. При интеграции с другими системами, позволяет изменять информацию в одном месте, а изменения произойдут на всех ресурсах размещения гостиниц, использующих channel manager.

2) *Авиабилеты*. Модуль отображает пользователю список имеющихся авиабилетов и их параметры. Для отображения авиабилетов используются входные данные получаемые от аэропортов. При выборе пользователем авиабилета, отсылается заявка на бронирование в аэропорт, затем пользователю приходит уведомление о состоянии бронирования.

Для работы с авиабилетами будет использоваться API Amadeus.

3) *Такси*. Пользователь выбирает маршрут от точки отправления до точки доставки, время и дату отправления, после он получит уведомления о состоянии заказа. Для формирования заказа на такси данные заявки заполненной пользователем будут отсылаться в службу такси. Для реализации будут использованы API лидирующих служб такси, такие как Uber, Yandex Taxi.

4) *Поиск*. Поиск будет выполняться из имеющейся базы данных. Так же в поиске пользователю будет предлагаться несколько стран, в которые отправились другие пользователи с того же региона. Для осуществления этой возможности, будут использоваться данные геолокации, полученные с помощью API Yandex maps или API Google maps.

5) *Профиль*. Данный модуль предоставляет следующие возможности:

- *Настройки*. Пользователь может изменять свои данные, такие как: имя, фамилию, пароль, почту, номер телефона и т.д.

- *Избранное*. Понравившиеся отели пользователь может добавить в избранное и посмотреть из в любое время данном разделе.

- *Отзывы*. Пользователь выставивший своё жильё в аренду может посмотреть отзывы людей, воспользовавшихся его услугами.

- *Помощь*. Пользователь может задать интересующие его вопросы по сайту модератору.

- История клиента. В данном разделе будет отображаться вся история путешествий клиента, она необходима для формирования возможных интересов пользователя, на основе которых будут предлагаться похожие услуги.

6) *Оплата*. Модуль, позволяющий пользователю производить оплату оказанных ему услуг в режиме online. Для реализации данного модуля будут использованы API: WebMoney, PayPal, Yandex Money. В дальнейшем будут добавлены другие системы для оплаты, требуемые пользователями.

7) *Регистрация нового объекта*. В данном модуле пользователь может зарегистрировать свой объект для сдачи его другим пользователям. Пользователь при регистрации указывает: тип и фотографии объекта, адрес, количество одновременно присутствующих человек, посуточная стоимость жилья, количество дней без платного снятия бронирования, а также предоставляемые дополнительные услуги.

Если пользователь использует Channel manager, он может найти его из списка для учета объекта. Если в списке не найден Channel manager используемый пользователем, будут разрабатываться услуги, позволяющие работать с максимальным их количеством.

Если пользователь зарегистрировавший свой объект не использует channel manager, ему будет предложен разработанный аналог channel manager в системе.

8) *Отчеты*. Пользователи выставляющие своё жилье в аренду, смогут сформировать отчет по своему объекту. Данная функция будет иметь более расширенные параметры для менеджера WEB-системы. Будут доступны детальные отчеты по работе всей WEB-системы. Для формирования отчетности будут использованы возможности фреймворка Laravel.

Выводы

Приведенная технология бронирования туристических услуг показывает, что при построении подобного рода Web-систем целесообразно использовать различные существующие глобальные и альтернативные системы бронирования услуг, такие как Amadeus, Galileo, Worldspan, Сирена-2 и т.п.

Сравнение функциональности аналогичных программ, представленных на рынке, показывает, что для реализации полной функциональности проектируемой Web-системы необходима разработка собственного туристического Веб-портала. Таким образом, была разработана функциональная модель Web-системы, выбраны средства реализации будущей системы, разработана структура модулей программной системы, соответствующей приведенной выше функциональной модели.

Литература

1. Всемирный день туризма/ [Electronic resource] / Интернет-ресурс. – Режим доступа: www/news.un.org/ru/story/2015/09/1271491- Загл. с экрана.
2. №88 Системы бронирования. Продажи в GDS/ADS. / [Electronic resource] / Интернет-ресурс. – Режим доступа: <https://infopedia.su/>- Загл. с экрана.
3. Светличная, В.А. Разработка функциональной структуры логистической системы формирования заказов для интернета-магазина / В.А. Светличная, Н.К. Андриевская, К.Ю. Чаленко // Информатика и кибернетика. -Д.: ДонНТУ, 2017. № 3(9). - С. 111-118.
4. Горничная, В.А. Разработка моделей для системы поддержки принятия решений при финансовой реструктуризации промышленного предприятия / В.А. Горничная, О.О. Шумаева, Н.К. Андриевска // Вісник Хмельницького національного університету. Економічні науки. 2013. № 4-1 (202). - С. 33-38.
5. PHP / [Electronic resource] / Интернет-ресурс. – Режим доступа: [www/ URL: https://ru.wikipedia.org/wiki/PHP](http://www/URL:https://ru.wikipedia.org/wiki/PHP) - Загл. с экрана.
6. MySQL / [Electronic resource] / Интернет-ресурс. – Режим доступа: [www/ URL: https://ru.wikipedia.org/wiki/MySQL](http://www/URL:https://ru.wikipedia.org/wiki/MySQL)- Загл. с экрана.

7. HTML / [Electronic resource] / Интернет-ресурс. – Режим доступа: www/ URL: <https://ru.wikipedia.org/wiki/HTML>- Загл. с экрана.

8. CSS / [Electronic resource] / Интернет-ресурс. – Режим доступа: www/ URL: <https://ru.wikipedia.org/wiki/CSS>- Загл. с экрана.

9. Laravel / [Electronic resource] / Интернет-ресурс. – Режим доступа: www/ URL: <https://ru.wikipedia.org/wiki/Laravel>- Загл. с экрана.

УДК 004.93.14

АВТОМАТИЗИРОВАННАЯ СИСТЕМА КОНТРОЛЯ И УПРАВЛЕНИЯ ДОСТУПОМ НА ОСНОВЕ АНАЛИЗА БИОМЕТРИЧЕСКИХ ПАРАМЕТРОВ ЧЕЛОВЕКА

Антипов Р.С., Мартыненко Т.В.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: 380713347428@yandex.ru, tatyana.v.martynenko@gmail.com

Аннотация:

Антипов Р.С., Мартыненко Т.В. Автоматизированная система контроля и управления доступом на основе анализа биометрических параметров человека. Приведено описание методов обработки изображений, используемых при идентификации человека, для повышения уровня достоверности биометрических систем. Рассмотрены все этапы решения поставленной задачи. Произведен сравнительный анализ существующих групп методов для решения поставленной задачи, а также выявлены основные проблемы, возникающие при обработке видеоданных.

Annotation:

Antipov R.S., Martynenko T.V. Automated access control system based on the analysis of human biometric parameters. A description is given of image processing methods used in human identification to increase the level of confidence in biometric systems. Considered all the stages of solving the problem. A comparative analysis of the existing groups of methods to solve the problem, as well as the main problems encountered in the processing of video data.

Введение

Биометрические данные используются для идентификации или различения людей на основе их уникальных особенностей. Исторически сложилось так, что основное применение биометрии было в уголовных расследованиях, и первоначально методы были довольно простыми. Измерения основывались на легко видимых особенностях тела, таких как шрамы, родимые пятна или расстояния между отдельными частями тела. Несмотря на простоту (хотя и трудоемкость) выполнения, эти методы страдали от неточных измерений и нечетких характеристик, которые вызывали более высокий риск неудачной идентификации или ложных срабатываний. В конце XIX века потенциал отпечатков пальцев как биометрического был признан. Это была гораздо более тонкая особенность, но с гораздо большей отчетливостью, чем ее предшественники. Система распознавания отпечатков пальцев вскоре стала наиболее часто используемой в англоязычных странах.

В настоящее время существует очень большое количество систем идентификации, основанных на различных типах биометрии. Они включают в себя распознавание радужной оболочки, отпечатков пальцев, сетчатки, голоса, лица и сосудов ладони. Последняя

технология представляет собой довольно новую технологию в биометрической области, которая постепенно внедряется во всем мире. Идея использования сосудистого паттерна кисти впервые была рассмотрена в начале 1990-х годов, но только в начале 2000-х был разработан коммерческий продукт, ставший популярным, когда было создано приложение для идентификации личности на основе рисунка вен на тыльной стороне ладони.

Хотя биометрия по-прежнему является важным инструментом уголовного расследования, в настоящее время она также используется в коммерческих продуктах, требующих аутентификации пользователей, таких как контроль доступа. Другое возможное применение биометрии – в наблюдениях, где используется распознавание лиц. Это область биометрии, которая была широко исследована, особенно после атак 9/11 [7][8].

Постановка задачи

Проблеме идентификации человека посвящено множество работ, однако в целом она ещё далека от разрешения. Основные трудности заключаются в том, чтобы можно было идентифицировать человека по изображению его сосудистого русла независимо от изменения ракурса и условий освещённости при съёмке.

Цифровое изображение, получаемое видеокамерой в момент времени t и имеющее по вертикали h , а по горизонтали w пикселей, обозначим $I_t(h, w)$. Видеопоток – это последовательность цифровых изображений (кадров) $I_t(h, w), I_{t+1}(h, w), \dots, I_{t+k}(h, w)$. Под прямоугольной областью интересов $ROI(x, y)$ понимается множество пикселей цифрового изображения $I_t(h, w)$, очерчивающих искомый объект, содержащее x пикселей по вертикали и y по горизонтали. Обнаружением объекта называется выделение области $ROI(x, y)$ на цифровом изображении $I_t(h, w)$ в момент времени t .

Цель исследования – обзор методов идентификации человека по изображениям для повышения уровня достоверности распознавания бесконтактных систем.

Решение задачи состоит из нескольких шагов. На рисунке 1 представлены основные этапы обработки изображений.

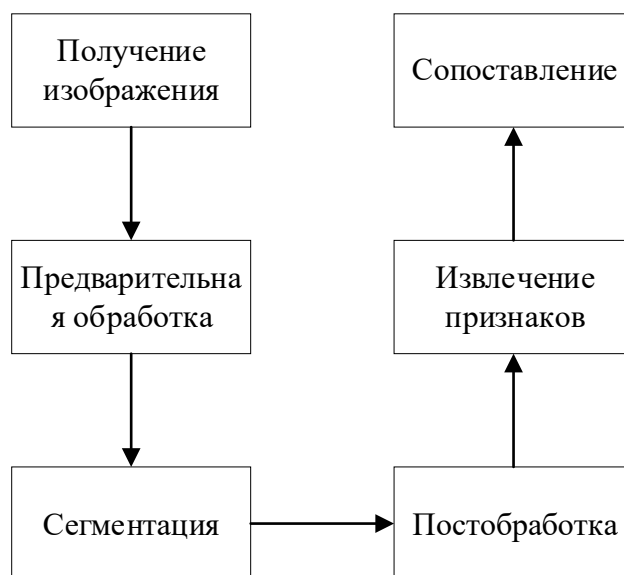


Рисунок 1 – Основные этапы обработки изображений

Обнаружение и распознавание объектов в кадре

В первую очередь после получения очередного кадра требуется выделить на нем объекты, представляющие интерес.

Проблема с выделением области интересов (ROI) заключается в том, что извлеченные области должны быть одного размера и с одинаковым расположением для каждого снимка. Для решения этой проблемы используются два метода извлечения ROI.

Первый метод предполагает фиксацию руки в определенном положении сразу под камерой [2].

Второй метод основан на извлечении информации, взятой из захваченного изображения. Для этого используются ключевые точки контура руки [6].

Удаление ненужной информации

Выделенное ROI содержит помимо рисунка вен много ненужной информации, от которой необходимо избавиться. Это выполняется в три этапа: предобработка, сегментация, постобработка.

Предобработка уменьшает шум в изображении, который мог возникнуть в результате работы некачественной камеры или наличием волосков. Для этого этапа применяется фильтр высоких частот (ФВЧ), а затем бинаризацию на основе гистограммы [3] либо фильтр низких частот (ФНЧ) Гаусса [2]. Затем для удаления 'шума' от волосков на руке применяется медианный фильтр.

После уменьшения шума изображение проходит **сегментацию**. Она используется для получения хорошего двоичного представления рисунка вен. Распространенным методом является локальное пороговое значение [1][2][4][5]. Это дает возможность получить хорошее разделение рисунка венозного русла от фона. Другие методы – направленный сосудистый рисунок [3] и обнаружение края [6] – являются более сложными в реализации.

Постобработка нужна из-за изменений диаметров вен, вызванных различными факторами, такими как температура окружающей среды, физические упражнения и т.д. Применяется также для изоляции узора вен, удаляя 'капли', не являющиеся частью рисунка вен.

При постобработке применяются следующие морфологические методы: дилатация (расширение, наращивание), эрозия (сужение), открытие (размыкание) и закрытие (замыкание).

Извлечение признаков и распознавание

Последние шаги, связанные с системой распознавания образов вен – это функция извлечения и распознавания. Эти шаги сильно зависят друг от друга, поскольку один метод распознавания обычно работает только с определенной характерной чертой. По этой причине были выбраны функции и соответствующие методы извлечения на основе методов распознавания.

Для преодоления проблемы нестабильности признака от окружающей среды, системе необходимо проанализировать общую форму рисунка сосудистого русла. Утоньшение – широко используемый метод для извлечения этой формы. Он преобразует рисунок вены в линию толщиной в один пиксель. Этот алгоритм часто модифицируется, чтобы избавиться от некоторых ненужных точек размером в один пиксель, а также для избавления от небольших ненужных ветвей [1][2].

Скелет, полученный этими методами, затем используется для извлечения признаков и сопоставления. Иногда в качестве признака используется сам скелет, иногда извлекаются такие признаки, как перекрестные и конечные точки.

При использовании скелета используется метод медиальной репрезентации оси элемента массива вен, а затем применение схемы «ограниченной последовательной корреляции» для сопоставления с шаблоном [1].

Для сопоставления точек чаще всего используют триангуляцию Делоне, модифицированное расстояние Хаусдорфа.

Чтобы быть эффективным, метод триангуляции Делоне требует, чтобы положение всех точек относительно друг друга было стабильным. К сожалению, в целом оказалось, что

это не относится к конечным точкам и точкам пересечения, извлеченным из рисунков утонченных вен. Из-за небольших различий в снятых изображениях данные точки, как правило, появляются случайным образом на утонченном изображении, в результате чего триангуляция сильно отличается даже между очень похожими изображениями. Данный метод имеет высокую эффективность при работе в контактных системах распознавания (отпечаток пальца).

Модифицированное расстояние Хаусдорфа (MPX) имеет низкое значение при сравнении выборок одного и того же шаблона и высокое у разных шаблонов. Это показывает, что MPX может использоваться для распознавания образов.

Для распознавания с помощью MPX требуется база данных извлеченных шаблонов, назначенная каждому зарегистрированному человеку. Когда неизвестный образец должен быть идентифицирован, значение MPX вычисляется между ним и всеми изображениями, соответствующими человеку в базе данных. После этого берется среднее значение этих расстояний, давая оценку того, насколько похож образец на шаблон этого человека. Это повторяется для всех зарегистрированных лиц. После этого неизвестный шаблон идентифицируется как принадлежащий к классу с самым низким MPX, который находится ниже заданного порога принятия решения. Если нет значения ниже порога принятия решения, то неизвестный шаблон классифицируется как неизвестный.

Производительность биометрических систем имеет важное значение для определения того, есть ли у системы потенциал для применения в реальных жизненных ситуациях. В таблице 1 представлены рассмотренные методы и их эффективность.

Таблица 1 – Эффективность методов

Название	Объекты	Изображения для тестирования	Попытки сопоставления		FAR	FRR	FTE
			Права доступа				
			Есть	Нет			
Кросс и Смит [1]	20	2	40	760	0%	7.5%	0%
Ванг и Лидхам [2]	12	6	72	792	0%	0%	0%
Университет Цинхуа [3]	13	5	260	3120	0%	4.6%	0%
Харбинский университет [3]	48	5	960	45120	0%	0.8%	0%
Миура (2004) [4]	678	1	678	459006	0.145%	0.145%	0%
Миура (2006) [5]	678	1	678	459006	1%	0%	0%
Лин и Фан [6]	32	15	480	14880	3.5%	1.5%	'малое число'

В таблице представлены три ключевых показателя производительности:

- 1) Частота ложного принятия (FAR) – вероятность того, что несанкционированное лицо принято в качестве уполномоченного лица.
- 2) Частота ложных отказов (FRR) – вероятность того, что уполномоченное лицо отклонено как неавторизованное лицо.

3) Отказ от регистрации (FTE) – вероятность того, что данный пользователь не сможет зарегистрироваться в биометрической системе из-за недостаточно отличительной биометрической выборки.

Данные показатели могут привести к неправильной оценке эффективности, если используются только два из них без третьего.

Обобщенная схема работы системы

Система контроля и управления доступом в упрощенном виде состоит из трех модулей:

1. Модуль считывания биометрических данных.
2. Модуль идентификации.
3. Модуль предоставления доступа.

Входной информацией для такой системы является видеопоток. Из него выделяется последовательность кадров, к каждому из которых применяется алгоритм детектирования руки. Получив область кадра с объектом, необходимо произвести сравнение ключевых признаков полученного объекта с имеющимися в БД. В результате чего, получаем отсылку на конкретного человека. В случае отсутствия совпадения, регистрируется попытка нарушения доступа. В случае идентификации личности выполняется проверка прав доступа.

Основным ограничением является тот факт, что система никогда не может измерить данную функцию с абсолютной точностью: хорошие измерения требуют точного оборудования и, как правило, объект должен быть расположен определенным образом для достижения полезных результатов. Последствия сбоя системы будут зависеть от приложения. Если пользователь ложно отклонен системой безопасности, ущерб обычно ограничен, поскольку пользователь может просто попробовать еще раз. Если, с другой стороны, самозванец ложно принят системой, потенциальный ущерб может быть большим. По этой причине высокая ложная скорость принятия (FAR) обычно более важна для системы безопасности, чем низкая истинная скорость принятия (TAR).

Выводы

В статье исследуется процесс идентификации человека на основе биометрического изображения. Проведен сравнительный анализ существующих категорий методов решения поставленной задачи. Определены направления в решении задачи. В процессе исследования предметной области были выявлены существенные проблемы, которые могут возникнуть при обработке потока видеоданных, а именно сложность идентификации человека при изменении ракурса съемки признака.

Литература

1. J. Cross and C. Smith, "Thermographic imaging of the subcutaneous vascular network of the back of the hand for biometric identification," in Security technology, 1995. Proceedings. Institute of Electrical and Electronics Engineers 29th Annual 1995 International Carnahan Conference on. Sanderstead, UK: IEEE, 1995.
2. L. Wang and G. Leedham, "A thermal hand vein pattern verification system," Lecture Notes in Computer Science, pp. 58-65, 2005.
3. Y. Ding, D. Zhuang, and K. Wang, "A study of hand vein recognition method," in International Conference on Mechatronic and Automation. Niagara Falls, Canada: IEEE, 2005.
4. A. Nagasaka, T. Miyatake, and N. Miura, "Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification," Machine Vision and Applications, vol. 15, pp. 194 - 203, 2004.
5. N. Miura, A. Nagasaka, and T. Miyatake, "Personal identification device and method," US Patent #2005/0 047 632A1, 2005.
6. K. Fan, C.-L. Lin, and W.-L. Lee, "A study of hand vein recognition method," in 16th IPPR Conference on Computer Vision, Graphics and Image Processing. Kinmen, ROC: IPPR, August 2003.

7. T. Frank. (2007, May) Face recognition next in terror fight. Internet. [Online]. Режим доступа: <http://www.usatoday.com/news/washington/2007-05-10-facial-recognition-terrorism-N.htm>
8. M. Kane. Face recognition grew even before 9/11. [Online]. Режим доступа: <http://news.cnet.com/2100-1023-883730.html>

УДК 004.932

ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА ДЛЯ КОМПЬЮТЕРНОЙ ОБРАБОТКИ ИЗОБРАЖЕНИЙ

Кисловская Я. О., Землянская С. Ю.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

Email: kislovskaya.yana@mail.ru

Аннотация

Кисловская Я. О., Землянская С. Ю. Инструментальные средства для компьютерной обработки изображений. В статье рассмотрены критерии оценки качества изображения. Исследованы существующие инструментальные средства для обработки изображений, а также предложена технология создания нового интегрального средства обработки, которое обрабатывает изображение в соответствии с предпочтениями пользователя.

Annotation

Kislovskaya Y. O., Zemlyansky S. Y. Tools for computer image processing. In article criteria for evaluation of quality of the image are considered. The existing tools for processing of images are investigated and also the technology of creation of new integrated tool of processing which processes the image according to preferences of the user is offered.

Постановка проблемы

Компьютерная обработка изображений актуальна в любой области деятельности.

При компьютерной обработке изображений решаются такие задачи, как улучшение качества изображений, измерение параметров, распознавание изображений, сжатие изображений, а также фильтрация и восстановление изображений.

Некоторые люди, которые не умеют на профессиональном уровне пользоваться программным обеспечением, обычно затрудняются с выбором фоторедакторов или нужного метода обработки изображения таким образом, чтоб изображение в результате было визуально приятным.

Кроме того, само понятие приемлемости (качественности) изображения – часто довольно субъективно и значение критериев оценки могут существенно отличаться у разных пользователей.

Поэтому удобно было бы иметь систему, которая на основании анализа изображений, которые пользователь считает качественными, вычисляла бы усредненные значения критериев качества изображения для этого пользователя и в соответствии с полученными значениями критериев качества выполняла бы предварительную обработку загруженного изображения. При этом можно оставить пользователю возможность «доводки» изображения, если такая необходимость возникнет.

Для того, чтобы разработать подобную систему, необходимо сформулировать критерии, по которым оценивается качество изображений и исследовать доступные инструменты обработки, которые могут быть использованы в подобной системе.

Обзор исследований по теме

Цифровое изображение $A(m,n)$ представлено в дискретном двумерном пространстве, где m – номер строки, n – номер столбца, а элемент на пересечении m -ой строки и n -го столбца называется пиксель, содержащий в себе код, обычно идентифицирующий цвет [1].

Существуют типы изображений с различной глубиной цвета: черно-белые штриховые, в оттенках серого (полутоновые), с индексированным цветом, полноцветные. Так, исходя из этого, существует большое количество форматов таких изображений: bmp, tiff, gif, psd, png, jpeg, ico, raw.

Формат JPEG (Joint Photographic Experts Group) предназначен для сохранения растровых файлов со сжатием. Для применения различных методов обработки в данной работе будут использоваться цифровые растровые изображения различных типов формата jpeg, так как они обеспечивают высокую точность передачи градаций цветов и полутонов.

Обработка изображения, основанная на обработке соответствующих сигналов, предусматривает внесение в формируемое изображение тех или иных изменений по сравнению с оригинальным изображением, а именно:

1. Изменение (искажение) изображения с целью достижения каких-либо эффектов (художественное улучшение);
2. Визуальное (заметное глазом) улучшение качества изображения (коррекция яркости и контраста, цветокоррекция и т.п.);
3. Объективное улучшение качества изображения (устранение искажений типа дисторсия, смаз, расфокусировка, увеличение резкости и т.п.);
4. Оконтуривание изображений – определение отдельных образов, выделение контуров изображения.

Применение методов обработки влияет на параметры изображения:

1. Яркость – представляет собой характеристику, определяющую то, насколько сильно цвета изображения отличаются от черного. Изменение яркости выполняется путем уменьшения или увеличения значения b каждого пиксела на одну и ту же величину, что приводит к тому, что все элементы корректируемого изображения становятся более светлыми или более темными.
2. Контраст – это характеристика изображения различия насыщенности цвета его наиболее ярких и наиболее темных участков.
3. Резкость – это степень размытости границы между соседними участками изображения с разной оптической плотностью (яркостью).

Возможны два подхода к оценке качества изображений: количественная оценка с помощью математических методов и субъективная оценка на основе экспертных оценок. Кроме этого они могут быть абсолютными или сравнительными [2].

Во всех рассматриваемых критериях оценивания качество изображений определяется числом, которое вычисляется по исходному изображению. Для расчета оценок используются следующие количественные критерии качества:

1. Для оценки яркости изображения применяется корректное построение гистограммы распределения яркостей элементов изображения.

Гистограммой цифрового изображения с уровнями яркости в диапазоне $[0, L-1]$ является дискретная функция $h(r_k) = n_k$, где r_k есть k -й уровень яркости, а n_k – число пикселей на изображении, имеющих яркость r_k .

Значения нормализованной гистограммы вычисляются по формуле:

$$p(r_k) = \frac{n_k}{n} \quad (1)$$

где $k = 0, 1, \dots, L-1$.

$p(r_k)$ - оценка вероятности появления пикселя со значением яркости r_k

По горизонтальной оси гистограммы отложены значения уровней яркости r_k , а по вертикальной – значения гистограммы $h(r_k) = n_k$ или $p(r_k) = n_k/n$, если они нормализованы.

Изображение, распределение значений элементов которого близко к равномерному и занимает весь диапазон возможных значений яркостей, будет выглядеть высококонтрастным и будет содержать большое количество полутонов.

2. Для оценки контрастности наиболее универсально используется отношение среднеквадратического отклонения к максимально возможному значению яркости.

$$C = \frac{2\sigma}{Y_{\max}} \rightarrow 1 \quad (2)$$

где σ - среднеквадратическое отклонение,

Y_{\max} - максимально возможное значение яркости.

C изменяется в диапазоне $0 \leq C \leq 1$, а его значение должно стремиться к 1 для наилучшего повышения визуального восприятия. Значение 0 соответствует однотонному изображению, значение 1 — максимально контрастному.

3. Для оценки резкости изображения вычисляется мера эксцесса двумерного спектра Фурье.

Алгоритм определения резкости изображения.

1. Выделить прямоугольную область на исходном изображении.

2. Выделенную область интереса преобразовать в частотную область.

3. Из преобразованной полученной области вычислить меру эксцесса двумерного спектра Фурье с помощью формулы:

$$\gamma_2 = \frac{\mu_4}{\mu_2^2} = \frac{\mu_4}{\sigma^4} \rightarrow \max \quad (3)$$

где μ_2 – второй центральный момент, μ_4 – четвертый центральный момент, σ - стандартное отклонение.

Эксцесс определяется как отношение четвертого центрального момента и квадрата второго центрального момента (стандартное отклонение в четвертой степени). Результат вычисления должен стремиться к максимальному значению, так как более резкому изображению соответствует большая величина эксцесса.

Обзор существующих систем и библиотек

Для обработки цифровых изображений существует большое количество инструментальных средств, например, онлайн фоторедакторы, прикладные программы, а также некоторые технологии и открытый исходный код для создания новых программных продуктов.

Все инструментальные средства дают возможность применить различные методы обработки изображений, такие как: контрастность, устранение шумов и размытие, регулировка яркости и насыщенности, увеличение резкости и четкости, изменение размера изображения, наложение фильтров и многое другое.

Также есть несколько вариантов для интеграции со сторонними системами [3]:

1. Взаимодействие с помощью SDK

Естественным развитием первого описанного подхода к интеграции стала разработка многочисленных SDK. SDK – комплект средств разработки, используемый разработчиками программного обеспечения. В состав этого комплекта входит набор полезных утилит, исходные коды и библиотеки.

Большинство методов обработки цифровых изображений уже разработаны и оформлены в виде различных библиотек с открытым исходным кодом, таких как: MatLab, OpenCV, JAI, ImageMagick, CamanJS, VTK, VIPS и многие другие, которые можно отнести к варианту интеграции с помощью SDK.

Все необходимые методы описаны в dll-файлах, которые необходимо подключать к создаваемому приложению, как и любую другую библиотеку. После этого нужно создать интерфейс для пользователя, который позволит обратиться к функциям с содержащимися в ней параметрами.

Далее рассмотрим подробнее некоторые из перечисленных выше библиотек.

OpenCV (*Open Source Computer Vision Library*) является библиотекой различных алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения с открытым кодом. Реализована на C#/C++, также разрабатывается для Python, Java, Ruby, Matlab, Lua и других языков. Библиотека открывает доступ к различным методам цифровой обработки изображений, таким как: считывание и вывод изображения, преобразование в различные цветовые модели, построение гистограммы яркости, выравнивание гистограммы, применение фильтров, сегментация изображений и др.

MatLab – это высокоуровневый язык и интерактивная среда для программирования, численных расчетов и визуализации результатов. Для него имеется возможность создавать специальные наборы инструментов, расширяющие его функциональность. Один из таких инструментов – пакет Image Processing Toolbox для моделирования и исследования методов обработки изображений.

ImageMagick – это кроссплатформенный консольный редактор изображений. Поддерживает множество графических форматов. Данное программное обеспечение доступно как в открытом коде, так и в виде API на многих языках программирования. В их число входят самые разнообразные интерпретаторы и компиляторы, начиная от PHP, Perl и до Java, Python и C++.

2. Взаимодействие через API

Следующий вариант программной интеграции – реализация производителем набора программных запросов (методов или точек входа), позволяющих взаимодействовать с программными и аппаратными средствами в формализованном виде, – API. Фактически это набор готовых процедур (библиотек, функций), предоставляемых для использования во внешних программных продуктах, легко подключаемых и используемых с любым современным языком программирования. Обмен информацией происходит посредством вызова неких функций, что позволяет организовать динамический обмен данными между приложениями.

Технология API - это определенный инструмент интеграции, с помощью которого выполняется мгновенное подключение к серверам других ресурсов.

API подразделяют на типы:

- Удаленный вызов процедур (Remote Procedure Call – RPC)
- Простой протокол доступа к объектам (Simple Object Access Protocol – SOAP)
- Передача состояния представления (Representational State Transfer – REST)

Идея **вызова удалённых процедур** состоит в расширении механизма передачи управления и данных внутри программы, выполняющейся на одной машине, на передачу управления и данных через сеть.

SOAP (Simple Object Access Protocol) является стандартизированным протоколом передачи сообщений между клиентом и сервером. Обычно он используется совместно с

HTTP(S), но может работать и с другими протоколами прикладного уровня (например, SMTP и FTP). SOAP использует несколько языков разметки: XML, XSD, WSDL.

REST (RESTful) - это общие принципы организации взаимодействия приложения/сайта с сервером посредством протокола HTTP. Особенность REST в том, что сервер не запоминает состояние пользователя между запросами - в каждом запросе передаётся информация, идентифицирующая пользователя (например, token, полученный через OAuth-авторизацию) и все параметры, необходимые для выполнения операции.

Всё взаимодействие с сервером сводится к 4 операциям:

1. получение данных с сервера (обычно в формате JSON, или XML);
2. добавление новых данных на сервер;
3. модификация существующих данных на сервере;
4. удаление данных на сервере.

Операция получения данных не может приводить к изменению состояния сервера [4].

Для каждого типа операции используется свой метод HTTP-запроса: получение – GET, добавление – POST, модификация – PUT, удаление – DELETE.

Каждый раз, когда пользователь посещает какую-либо страницу в сети, он взаимодействует с API удалённого сервера. API — это составляющая часть сервера, которая получает запросы и отправляет ответы.

Когда компания предлагает своим пользователям API, это означает, что она создала ряд специальных URL, которые в качестве ответа возвращают только данные. Такие запросы часто можно отправлять через браузер. Так как передача данных по протоколу HTTP происходит в текстовом виде, браузер всегда сможет отобразить ответ.

Большинство онлайн сервисов предоставляют свой API для использования другими разработчиками бесплатно, но с ограничениями (например, доступ к методам на 30 дней или до 100 запросов в день). Для полноценного доступа необходима покупка подписки. В каждом из данных вариантов сервис выдает определенные ключи для доступа к готовым методам: открытый ключ (APP_ID) и закрытый ключ (KEY).

Закрытый ключ скрыт от других и используется для шифрования определенных параметров в запросе. Закрытый ключ можно изменить по необходимости. Открытый ключ открыто используется как идентификатор приложения. Это бесполезно без закрытого ключа. Открытый ключ может быть деактивирован, если он не использовался в течение длительного периода времени.

Для разработки программного модуля для улучшения качества изображения применение взаимодействия как с помощью SDK, так и с помощью API Rest является наиболее подходящим. Поэтому рассмотрим пример применения этих технологий.

Примером онлайн фоторедактора, который может предоставить свой API со всеми методами обработки является сервис «Editor.Pho.to». Среди методов, содержащихся в Pho.to API выделено несколько категорий:

1. Редактирование. В эту категорию входят методы для коррекции экспозиции, контрастности и цветовой температуры, регулировки яркости и насыщенности, устранения шумов и размытия, а также поворота, изменения размера и обрезки изображений.
2. Ретушь лица. Этот набор методов для автоматической ретуши портрета (устранения эффекта «красных глаз», устранения дефектов кожи).
3. Фильтры. Эта категория включает в себя различные художественные фильтры и эффекты, такие как цветовые эффекты (сепия, черно-белый); старинные фотофильтры; различные художественные и живописные эффекты (мел и уголь, эскиз) и набор других фильтров.
4. Коллаж. Это оболочка, которая позволяет использовать любую комбинацию других методов внутри себя.

5. Анимация

6. Карикатура. Это универсальный метод для изменения выражения лица на портретных фотографиях.

7. Распознавание лиц.

В приведенном ниже примере запроса вызывается метод обесцвечивания, применяемый к конкретному изображению, которое указывает пользователь.

```
<image_process_call>
  <image_url>http://developers.pho.to/img/girl.jpg</image_url>
  <methods_list>
    <method order="1">
      <name>desaturation</name>
    </method>
  </methods_list>
</image_process_call>
```

Далее приходит ответ на запрос с информацией об успешности или неудаче выполнения.

```
<image_process_response>
  <request_id>REQUEST_ID</request_id>
  <status>OK</status>
  <description>DESCRIPTION</description>
  <err_code>0</err_code>
</image_process_response>
```

После вызова метода необходимо получить результирующее изображение с помощью приведенного ниже действия.

http://opeapi.ws.pho.to/getresult?request_id=:REQUEST_ID

Так же, как и после первого запроса, необходимо получить ответ об успешности и обесцвеченное изображение.

Таким образом, применяя методы обработки изображения данного сервиса, пользователь может оценить визуально, какому результату он отдаст предпочтение.

Используя вышеописанный подход, в разрабатываемой системе будут реализованы такие методы обработки: коррекция экспозиции и контрастности, регулировки яркости и насыщенности, устранения шумов и размытия. Дополнительно с помощью подключаемых библиотек PhotoEditor SDK, CamanJS и PSX будет также реализовано применение различных фильтров и некоторых аналогичных методов, используемых через API.

Кроме этого будет сформирована библиотека различных изображений, обработанных с разным качеством. Пользователь получит возможность выбрать то изображение, которое наиболее соответствует его представлениям о качественном изображении. Затем он сможет загрузить собственное изображение для обработки, и оно будет обработано системой в соответствии с параметрами, которые установлены для выбранных пользователем изображений. Если же результат в чем-то не будет соответствовать предпочтениям пользователя, то он сможет откорректировать некоторые параметры при помощи инструментов обработки, предоставляемых разрабатываемой системой.

Выводы

Были исследованы различные инструментальные средства и технологии обработки изображений. А также рассмотрены критерии оценки качества изображения. Принято решение создать систему, которая будет интегрировать в себе различные функциональные подсистемы обработки изображений через технологию API, применяя различные методы обработки, также будут использованы обрабатывающие алгоритмы различных SDK библиотек для веб-разработки. Для взаимодействия с API планируется использование веб-разметки HTML и языка JavaScript. С помощью этой системы будет происходить

предварительное обучение на существующих изображениях и вычисление параметров изображения, предпочтительных для пользователя, после чего пользователю предоставится возможность загрузить свое изображение и далее выполнится обработка в соответствии с параметрами, полученными при обучении.

Литература

1. Компьютерная обработка изображений. [Электронный ресурс] – Режим доступа: URL: http://aco.ifmo.ru/el_books/image_processing/1_01.html
2. Ю.И. Мониц, В.В. Старовойтов «Оценки качества для анализа цифровых изображений» «Искусственный интеллект» 4'2008. Стр. 376-386.
3. Варианты протоколов для интеграции со сторонними системами в современных СКУД. [Электронный ресурс] – Режим доступа: URL: <http://www.techportal.ru/196160>
4. Введение в REST API. [Электронный ресурс] – Режим доступа: URL: <https://myrusakov.ru/rest-api-introduction.html>

УДК 004.42

РАЗРАБОТКА БАЗЫ ДАННЫХ АЛГОРИТМОВ

Павлов М.В., Харитонов Ю.Е.

Донецкий национальный технический университет
кафедра экономической кибернетики

E-mail: pavlovmark24@gmail.com

Аннотация:

Павлов М.В., Харитонов Ю.Е. Разработка базы данных алгоритмов. В данной работе рассмотрено создание информационной системы, главной задачей которой является хранение и предоставление доступа к информации о различных алгоритмах. На данном примере продемонстрирована разработка системы база данных – сервер – тонкий клиент.

Annotation:

Pavlov M.V., Kharitonov, Yu.E. Development of a database of algorithms. This paper describes the creation of an information system, the main task of which is to store and provide access to information about various algorithms. This example demonstrates the development of a database - server - client system.

Введение

Сегодня в сети Интернет встречаются множество полезных и интересных инструментов, позволяющих решать различные задачи. Такие сайты заметно отличаются от обычных информационных ресурсов. Такой чертой является взаимодействие с пользователем, проявляющееся в запросе и обработке данных, которые передает клиент, для решения конкретной задачи. От их указания может зависеть содержимое страниц. Другим признаком можно выделить динамичность: получение актуальных данных без необходимости перезагрузки страницы. Вышеуказанные положения и составляют понятие веб-приложения.

Разработка веб-приложений представляет собой процесс, включающий в себя множество этапов. Их количество и содержание зависит от конкретной архитектуры. Рассмотрим вариант на примере разработки базы знаний по алгоритмам. [1]

Пройдем по всей архитектуре снизу вверх (рис. 1), начиная с базы данных. В ее задачи будет входить хранение данных, а также их предоставление по запросу от сервера. Учитывая особенности предметной области, классическая реляционная модель хранения данных

полностью удовлетворяет все потребности системы, поэтому средством общения сервера и базы данных будет выступать структурный язык запросов (SQL) [2]. В конкретной реализации используется СУБД H2, отличающаяся широким функционалом при компактном размере и нетребовательностью к ресурсам.



Рисунок 1. Архитектура разработанного приложения.

Переходим на серверную часть приложения. Она обслуживает клиентов, принимая от них запросы, отвечает на них, при этом подключается к базе данных для синхронизации состояния.

Сервер делится на три слоя, каждый из которых выполняет определенный набор функций. Охарактеризуем их, указывая практическое применение для нашей работы.

Первый слой представляет собой интерфейс для работы с данными, при этом он построен таким образом, чтобы возможное переключение на другую базу данных не оказывало никакого влияния на другие части приложения. Используя данный подход, этот слой называют DAO (Data Access Object) или уровень класса.

Работа с каждой из сущностей предметной области осуществляется за счет создания специального репозитория, реализующего соответствующий DAO и имеющий набор методов для выборки, создания, модификации или удаления тех или иных данных.

Второй слой представляет собой набор сервисов. Они представляют основную бизнес-логику приложения. Именно здесь обрабатываются приходящие от пользователей запросы. Каждый из сервисов выполняет отдельную задачу, при этом ему предоставляется доступ только к тем репозиториям, которые действительно нужны.

Создаваемое приложение будет иметь следующие сервисы: авторизация, регистрация, взаимодействие с каталогом, получение информации по отдельному алгоритму, работа с источниками, администрирование и другие.

Следующий уровень имеет название слоя контроллеров. Здесь происходит первичная обработка запросов от клиентов. Когда он обращается к серверу, вызывая определенный метод API, запрос приходит к специальному диспетчеру, имеющему информацию в виде карты, в которой каждому существующему пути соответствует метод какого-либо контроллера, а также осуществляется проверка на доступность данного ресурса для пользователя.

С другой стороны каждый метод, включенный в API приложения, имеет возможность получать данные и отправлять их как результат запроса. В его реализации находятся вызовы отдельных сервисов, с передачей отдельной информации. В создаваемом приложении имеется множество контроллеров, как правило, совпадающих по имени с сервисами.

Таким образом, серверная часть играет важную роль в работе приложения. Для взаимодействия с клиентами предоставляется специальный API, в котором описаны все методы, их параметры, что будет возвращаться по завершению обработки запроса, а также его доступность.

Следующая часть приложения, через которую клиент и взаимодействует с сервером, то есть пользовательский интерфейс представляет собой небольшое веб-приложение, написанное на фреймворке Vue.js.

По структуре клиент разбивается на отдельные компоненты. Каждый из них имеет собственные поля, методы, в них обрабатываются события от пользователя, получают уведомления от других компонентов. Они могут вкладываться в друг друга, образуя различные иерархии [3].

Строение каждого компонента состоит из трех частей: шаблон, написанный на HTML и задающий общую структуру компонента, то есть как он будет отображаться; далее описываются его переменные состояния, методы, обработчики и т. д.; последняя часть, необязательная, описывает визуальную составляющую, может реализовываться как на чистом CSS, так и на любом препроцессоре.

Для работы с данными, а точнее их синхронизации между компонентами и сервером, используется специальное расширение — менеджер состояний Vuex. По сути оно представляет собой локальное хранилище, в которое загружаются данные с сервера, а затем по запросу раздаются компонентам. Для каждой сущности создается специальный класс, содержащий четыре раздела. Первый раздел (state) — описывает, каковая будет структура хранимых данных. Второй раздел (mutations) — задает набор модификаций, которые могут применяться к данным. Стоит отметить, что никаким другим образом изменить данные невозможно. Третий раздел (actions) — содержит описания событий, которые могут возникать при работе с данной сущностью. Рекомендуются вызывать мутации именно из-под событий, что и выполняется в данной работе. Четвертый раздел (getters) — определяет методы для получения той или иной информации по сущности, обладает свойством кэширования, то есть метод выполняется единожды, а его результат сохраняется и возвращается при каждом вызове геттера. Если же данные, от которых зависит результат изменились, происходит перерасчет метода. Таким образом, данное расширение координирует работу всех компонентов, предоставляя актуальную информацию.

При создании приложения был реализован подход, когда все находится в корневом компоненте, а остальные элементы, которые отображают различные части приложения, отображаются с помощью применения маршрутизации, где каждому компоненту в соответствие установлен определённый адрес.

Приведем примеры различных компонент.

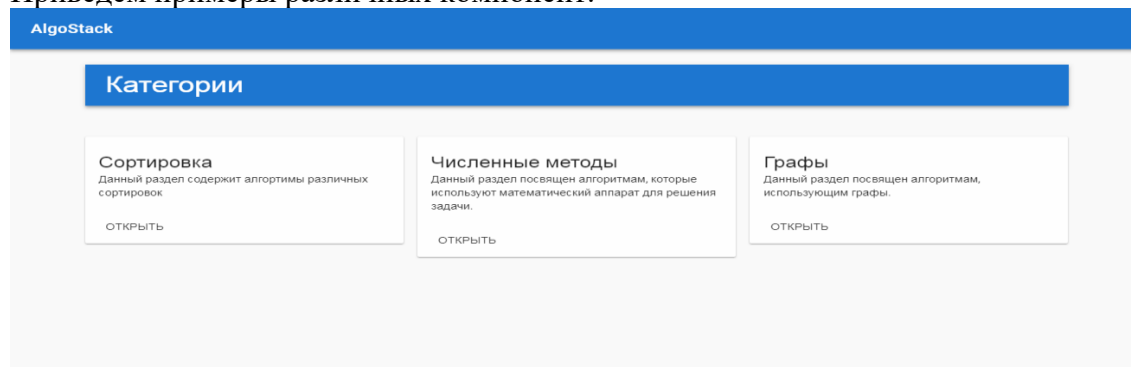


Рисунок 2. Домашняя страница для посетителя

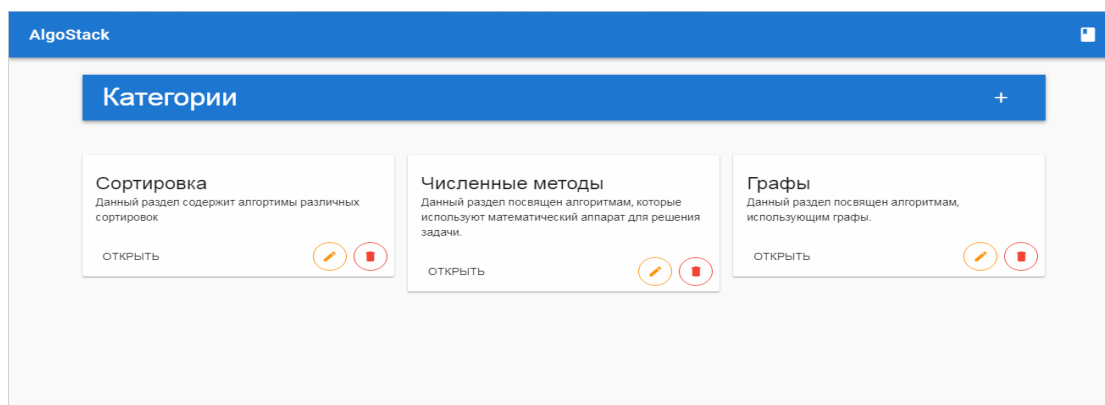


Рисунок 3. Домашняя страница для администратора.

Как видим в приложении также реализовано распределение ролей, согласно которому пользователь, имеющий те или иные привилегии имеет различный доступный функционал.

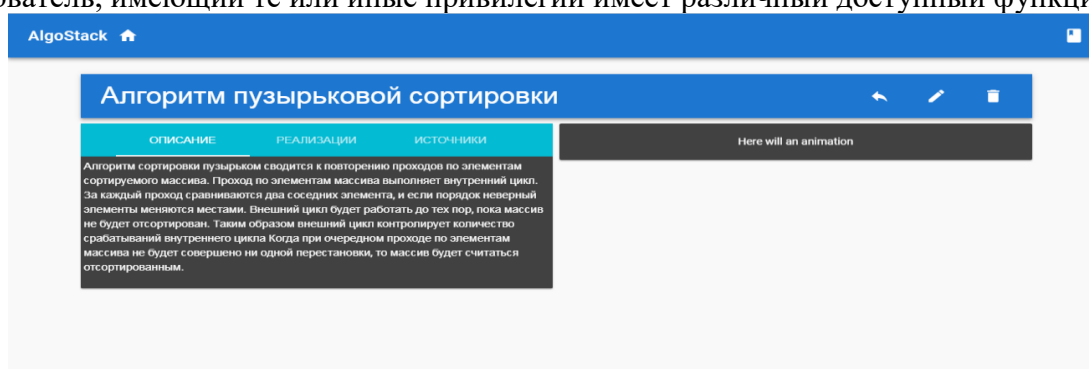


Рисунок 4. Просмотр информации по конкретному алгоритму.

Выводы

Таким образом, получили приложение, которое взаимодействует с базой данных, имеет функционал, разделенный на две роли: администратора и обычного пользователя. Серверная часть выполнена по архитектуре: DAO — сервисы — контроллеры, где каждый слой выполняет свою работу, начиная от подключения и взаимодействия с базой данных, заканчивая обработкой пользовательских запросов. Реализация клиентской части написана на фреймворке Vue.js в идеологии компонентного построения интерфейса, а также с наличием менеджера состояний приложения Vuex, что обеспечивает синхронизацию данных и динамическое взаимодействие с пользователем.

В перспективах развития данного приложения имеется много задач. Некоторые из них — это анимация работы алгоритмов, для демонстрации большей наглядности, возможность пошагового выполнения алгоритмов, а также пользовательские наборы данных для тестирования их работы наряду с сравнением между разными алгоритмами по заданным критериям.

Литература

1. Filipova O. Software Development From A to Z / Olga Filipova, Rui Vilão. - Berlin, Germany : Springer Science, 2018. - 299с.
2. Карпова И.П. Базы данных: Учебное пособие по курсу «Базы данных» - М., РИО МГИЭМ, 2009. - 118 с.
3. Callum Macrae Vue.js: Up and Running / Macrae Callum. - Sebastopol, USA : O'Reilly, 2018. 173с.

УДК 004.056

ПЛАГИАТ. НАРУШЕНИЕ И ЗАЩИТА АВТОРСКИХ ПРАВ.

Ищук Л.А.

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: Iyubkagolubka@gmail.com

Аннотация:

Ищук Л.А. Плагиа́т. Нарушение и защита авторских прав. Проблемы нарушений авторских прав сегодня очень актуальны. С появлением Интернета затруднилась борьба с плагиатом, так как обмен информацией стал проще. В статье рассмотрены проблемы авторских прав и основные способы их защиты. Описаны основные термины и выделены основные классификации плагиата.

Annotation:

Ishchuk L.A. Plagiarism. Copyright infringement and protection The problems of copyright violations are very relevant today. With the advent of the Internet, the fight against plagiarism became more difficult, as the exchange of information became easier. The article describes problems of copyright and the main ways of their protection. Also the basic terms and the main classifications of plagiarism are highlighted.

Введение

Самое раннее использование слова «плагиат» происходит от латинского слова «plagiarius» конца 1500-х годов. Слово «плагиат» переводится как «похититель». Римский поэт использовал этот термин, чтобы выразить свое разочарование тем, что другой человек похитил его работу. Примерно до XVII века писатели свободно пользовались произведениями других авторов.

Письменные слова и идеи считались общественной собственностью, а не собственностью одного человека. Было понятно, что для того, чтобы учиться, нужно сначала подражать. В конце концов, авторы настаивали на признании и упоминание своих оригинальных идей, таким образом, родилось определение плагиата.

В настоящее время, когда информация и существующие работы доступны, плагиат особенно преобладает. Теперь существует возможность не только легко исследовать почти все в интернете, но и просто скопировать чужие идеи и подписать их своим именем. В дополнение к словам, многие определения плагиата в настоящее время включают медиа, видео и изображения как интеллектуальную собственность, которая может быть плагиатом.

Авторские права – комплекс прав автора – правообладателя, закрепленных действующим законодательством и нацеленных на использование произведения и, кроме того, осуществление и защиту личных неимущественных авторских прав.

Объекты авторских прав:

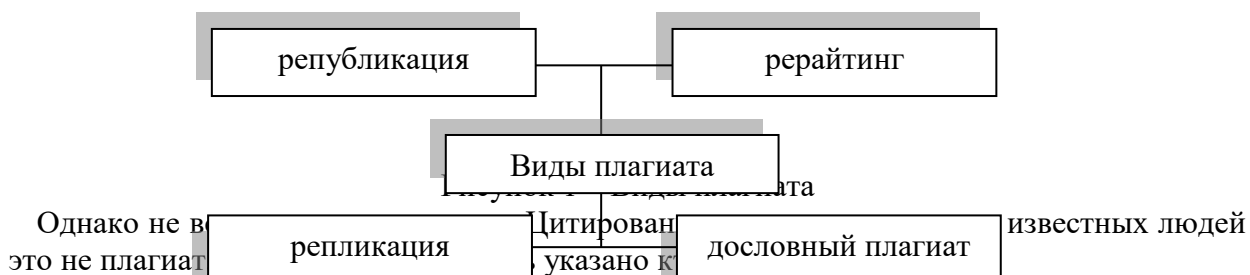
- литературные произведения; пр
- произведения графики, дизайна, живописи, скульптуры; фо
- торграфические произведения; ау
- аудиовизуальные произведения; му
- языковые произведения с текстом или без текста.

Плагиат

Плагиат это – акт присвоения или копирования чьей-либо художественной, письменной или другой творческой работы, идеи, исследования частично или полностью без указания источника или автора.

«Если вы крадете у одного автора, это плагиат; если вы крадете у многих - это исследование», - сказал Мизнер Уилсон.

Прежде всего, плагиат определяется как использование слов или идей других людей без четкого признания источника информации. Плагиат может принимать множество форм, таких как копирование всего источника и представление его как своего собственного, копирование разделов из источника без надлежащего подтверждения, перефразирование материала из источника без надлежащего подтверждения и покупка эссе у другого студента или с сайта.



Цитирование.

Цитирование – это способ, которым автор объясняет читателям, что определенное текстовое содержание, содержащее в конкретной статье взято из другого источника.

Если в статье используется такой способ, то обязательно должна содержаться информация об авторе:

- на
- звание статьи; но
- мера страниц, с которых взят материал; са
- йт, на котором документ можно посмотреть в открытом доступе.

Проблемы нарушения авторских прав

Проблема плагиата довольно широко распространилась в разных сферах жизни. Но уже сегодня ее не обходят стороной, а пытаются решать. Защита авторских прав принадлежит к числу наиболее сложных цивилистических проблем, имеющих непреходящую научную и практическую значимость.

Пользователи сети оправдывают нарушение авторских прав тем, что «так делают все». Таким образом, нарушения авторских прав в Интернете наносят большой моральный и экономический ущерб авторам и правообладателям.

Решение проблем нарушения авторских прав

В современном обществе производство и распространение информации изменилось радикально. Развитие технологий побудило правительство создать закон об авторском праве, в котором указано, что является допустимым и недопустимым при использовании материала, находящегося под защитой авторского права.

В соответствии со ст.1257 ГК РФ «Автором произведения науки, литературы или искусства признается гражданин, творческим трудом которого оно создано. Лицо, указанное в качестве автора на оригинале или экземпляре произведения, считается его автором, если не доказано иное.».

Законодательства других стран, подписавших Бернскую конвенцию об охране литературных и художественных произведений, имеют аналогичные нормы. Таким образом вводится понятие презумпции авторства и устанавливается заявительный характер авторского права.

Существует три основных способа защиты авторских прав в интернете:

- Отправление писем и претензий собственнику сайта или хостинг-провайдеру
- Судебный спор
- Взаимодействие с поисковыми сервисами для удаления страниц из результатов поиска

Выводы

В наше время проблема плагиата стала огромной или широко распространенной и присутствует практически во всех сферах человеческой деятельности, особенно в науке. В рамках данной статьи сформулированы основные проблемы нарушения авторских прав и пути их решения.

Литература

1. Жарова А.К. Правовая защита интеллектуальной собственности: учеб. пособие [для вузов] / под общ. ред. С.В. Мальцевой; Высш. шк. экономики. - М.: Юрайт, 2011. - 378 с.
2. Попов Р.М. Нарушение авторских прав публичным воспроизведением музыкальных произведений // Право и экономика. - 2011. - № 7. - С.45-49.
3. Примакова О.М. Нарушение авторского права в сети Интернет // Правовые вопросы связи. - 2011. - № 1. - С.32-35.
4. Щербак Н. Авторское право и наследование // Интеллектуальная собственность. Авторское право и смежные права. - 2011. - № 8. - С.26-29.
5. Васильева Т.В. О соблюдении авторских прав эпоху развития высоких технологий // Современное право. - 2011. - № 5. - С.102-104.
6. Ефимова Е.Г. Защита авторских и смежных прав на современном этапе // Патентное дело. - 2011. - № 5. - С.26-28.
7. Невская М.А., Тарасова Е.Н., Сухарев Е.Е. Авторское право в издательском бизнесе и СМИ: практ. пособие. - 2-е изд. - М.: Дашков и К, 2012. - 345 с.

УДК 004.932

ПОДСИСТЕМА МОНИТОРИНГА И КОНТРОЛЯ УСТАЛОСТИ ВОДИТЕЛЯ

Кожухов А.С., Васяева Т.А.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: artemt-34@mail.ru

Аннотация:

Кожухов А.С., Васяева Т.А. Подсистема мониторинга и контроля усталости водителя. В работе описана обобщенная схема работы подсистемы мониторинга и контроля усталости водителя. Приведен сравнительный анализ методов обнаружения и распознавания объектов в кадре для получения информации о состоянии водителя. Выявлены основные проблемы, возникающие при выполнении данной задачи.

Annotation:

Kozhukhov A.S., Vasyaeva T.A. Subsystem for monitoring and controlling driver fatigue. The article presents the methods of analytical processing of photos to obtain information about the status of the driver. The main stages of solving the problem were considered. A comparative analysis of the existing groups of methods was conducted. The main problems that arise when performing this task are identified.

Введение

На сегодняшний день, причиной каждого пятого дорожного происшествия становится сонливость. Многие водители за рулем автомобиля испытывают чувство усталости или сонливости, даже не подозревая о том, что находятся в таком состоянии. Понятие «вождение в полусонном состоянии», возможно, звучит несерьезно, но количество несчастных случаев, вызванных засыпанием водителя за рулем, с каждым годом уносит все больше жизней и приводит к травматизму. Вождение в состоянии сонливости настолько же опасно, как и вождение в состоянии алкогольного опьянения, о чем свидетельствует анализ числа травм и смертей, полученных при вождении. Отслеживание такого опасного состояния может помочь обратить внимание водителей на стиль своего вождения и связанные с этим риски, тем самым снизить процент неосторожного вождения и улучшить навыки безопасного поведения на дороге.

Системы автоматизированной поддержки водителя (Advanced Driver Assistance Systems, ADAS) нацелены на оказание помощи водителю в деле недопущения дорожно-транспортных происшествий (ДТП) или смягчения их последствий. Предупреждающие сигналы высокой приоритетности подаются этими системами для стимулирования бдительности и своевременных и надлежащих действий водителя в ситуациях, когда может иметь место опасность возникновения серьезных повреждений или гибели людей.

Все ADAS-системы можно условно разделить по форме представления на две категории:

- мобильные приложения, устанавливаемые на смартфон из магазина приложений (Google Play / App Store);
- использующие внешние камеры, сенсоры, датчики, чипы и т.д. вместе с программными средствами, встраиваемые в автомобили на заводах-производителях или устанавливаемые постфактум.

На рынке ADAS-систем существует немало количество решений от разных компаний. Можно отметить следующие недостатки интегрируемых ADAS-систем:

- стоимость таких систем остается достаточно высокой;
- они доступны в основном только в виде дополнительной опции для дорогих и эксклюзивных автомобилей.

Постановка проблемы

В работе рассматривается разработка подсистемы, которая бы предотвращала случаи засыпания водителя во время езды. Исходя из того, что далеко не каждый автомобиль оснащен бортовым компьютером, возникает необходимость в реализации именно портативной системы. Наиболее удобным вариантом будет система, реализованная на базе мобильного устройства. В процессе работы система будет делать фото лица водителя с определенной частотой, детектировать глаза и анализировать их «степень закрытости». Исходя из полученных данных, программа решит, стоит ли оповещать водителя об опасности.

Цель исследования – разработка подсистемы мониторинга и контроля усталости водителя на базе Android.

Обобщенная схема работы системы

Подсистема мониторинга и контроля усталости водителя в упрощенном виде состоит из трех модулей:

1. Модуль детектирования лица водителя в кадре.
2. Модуль распознавания глаз по ключевым признакам.
3. Модуль отслеживания лица и глаз на следующих кадрах, с последующим анализом.

Входной информацией для такой системы является последовательность фото, снятых с определенной частотой. К каждому из них применяется алгоритм детектирования лица водителя. Затем производится отслеживание на обнаруженном лице водителя положение глаз и проверяется на усталость с помощью имеющихся методов.

Решение задачи мониторинга состояния водителя состоит из нескольких шагов. На рисунке 1 представлена обобщенная схема работы.

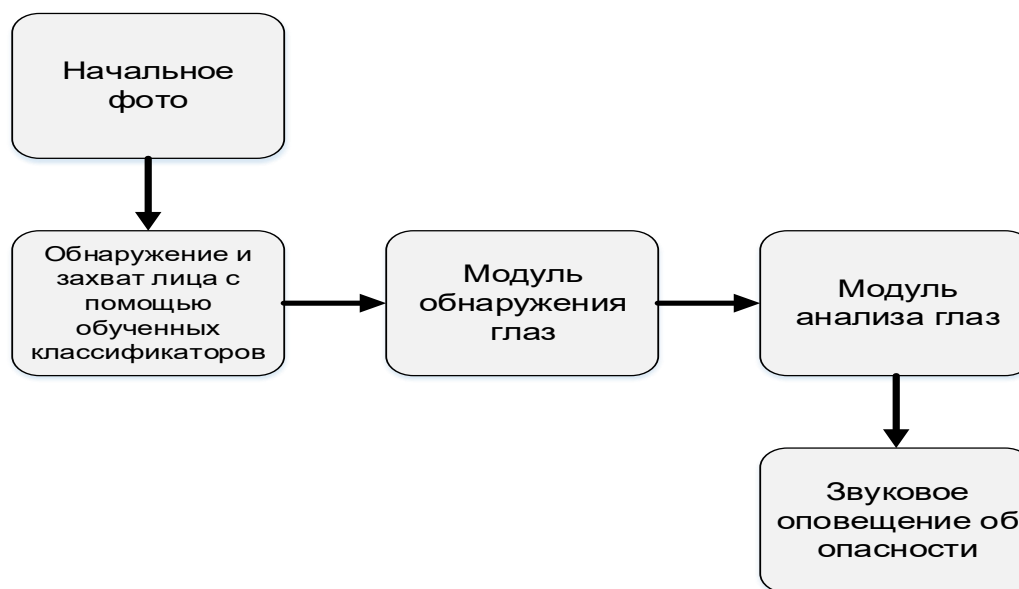


Рисунок 1 – Обобщенная схема работы подсистемы мониторинга и контроля усталости водителя

Обнаружение и распознавание объектов в кадре

В первую очередь после получения очередного кадра, требуется выделить на нем объекты, представляющие интерес, а именно лицо водителя.

Для решения данной задачи существует множество различных подходов, для дальнейшего анализа были выделены следующие группы методов: основанные на сопоставлении с шаблоном и основанные на извлечении признаков.

Детектирование объектов с использованием метода поиска объекта по шаблону

Данная группа методов работает по следующему принципу: требуется два компонента – исходное изображение и изображение-шаблон. Чтобы определить сходный участок, необходимо наложить шаблон на снимок и попиксельно двигать его, слева направо, сверху вниз. В каждом положении вычисляется метрика, которая показывает, совпадают картинки или нет. Для каждого положения шаблона метрика заносится в результирующую матрицу R. Каждое положение (x,y) в R содержит метрику совпадений.

После окончания сравнения лучшие совпадения находятся в глобальных минимумах (при использовании метода наименьших квадратов) или максимумах (при использовании метода корреляции или метода кросс-корреляции).

Данная группа методов хорошо применима при анализе сцен, в которых камера статична, а все экземпляры искомым объектов выглядят одинаково. На практике это приводит к следующим проблемам: в случае изменения ракурса съемки, перекрытия сразу наблюдается неустойчивая работа метода.

Категориальное распознавание объектов

Чтобы избежать вышеописанных проблем, было принято решение воспользоваться альтернативным подходом, а именно поиском лиц с помощью обученного классификатора. Построение классификатора подразумевает выделение некоторых характерных признаков на изображении, применение машинного обучения на наборе образцов, на которых заведомо присутствуют или отсутствуют нужные объекты, и потом применение классификатора к очередному изображению с целью поиска объектов на нем.

Среди наиболее популярных методов данной группы можно выделить следующие: метод Виолы-Джонса, главных компонент, преобразования Хафа. Сравнение методов хорошо описаны в [9].

Основные достоинства и недостатки работы каждого метода представлены в табл. 1.

Таблица 1 – Достоинства и недостатки методов группы категориального распознавания

Название методов	Достоинства	Недостатки
Метод главных компонент	<ul style="list-style-type: none"> - простота реализации; - низкое потребление памяти для хранения объектов; - простота добавления новых обученных фигур к уже обученному алгоритму. 	<ul style="list-style-type: none"> - очень чувствителен к освещенности; - чувствителен к углу поворота объекта; - задачей алгоритма не является распределение входных данных по классам.
Метод Виолы-Джонса	<ul style="list-style-type: none"> - обладает крайне низкой вероятностью ложного обнаружения; - высокая точность распознавания (даже при небольших поворотах фигуры); - высокая скорость работы. 	<ul style="list-style-type: none"> - обучение классификатора происходит очень медленно; - необходимо большое количество изображений искомого объекта для формирования обучающей выборки.

Преобразования Хафа	- работает в условиях перекрытий; - ищет все имеющиеся на изображении экземпляры объекта за один проход.	- сложность алгоритма растет экспоненциально с увеличением числа параметров модели.
---------------------	---	---

Для того, чтобы выяснить целесообразность применения описанных методов в системе, было произведено сравнение с точки зрения точности распознавания, времени обработки одного изображения, вычислительной сложности и объема обучающей выборки. Результаты сравнения представлены в таблице 2.

Таблица 2 – Сравнение методов обнаружения объекта

Метод	Объем обучающей выборки, шт.	Время обработки одного изображения, мс	Вычислительная сложность	Точность распознавания
Виолы-Джонса	14416	67	++	>90%
Главных компонент	24300	100	+	~90%
Преобразования Хафа	27159	100	++	~80%

Примечание: количеством знаков «+» отмечается степень различия для критерия.

Описание работы подсистемы

Данные с камеры смартфона, такие как положение лица, глаз и зрачков, анализируются в потоковом режиме. В таблице 3 приведена тестовая выборка из четырех экспериментов, представляющих координаты и размеры найденных областей объектов на изображении и временную оценку работы алгоритма. Значения переменных X и Y описывают координаты, а W и H – ширину и высоту найденного объекта соответственно. Координаты и размеры объектов на изображении X , Y , W , H представлены в единицах цифровых изображений, пикселях. Как можно заметить, значения как X и Y , так и W и H изменялись на протяжении экспериментов, и это можно объяснить тем, что найденный объект то приближался, то отдалялся в кадре, и соответственно менялось его расположение. Так, например, в эксперименте 3 лицо человека отдалялось. Время распознавания также незначительно менялось, и это можно связать с особенностями работы мобильной платформы Android и самим смартфоном, на котором было запущено данное приложение. Можно также отметить, что среднее время обработки одного кадра из видеоряда составляет 75 мс, что позволяет говорить о том, что система работает в реальном времени с точки зрения выполняемых задач пользователя в рамках проектируемого приложения. Данными расчетами занимались Лашков И.Б., Смирнов А.В., Кашевник А.М. [8].

Таблица 3 - Значения координат найденных объектов на изображении

Лицо				Левый глаз				Правый глаз				Левый зрачок				Правый зрачок				Время, мс
X	Y	W	H	X	Y	W	H	X	Y	W	H	X	Y	W	H	X	Y	W	H	
243	108	372	372	429	190	163	124	266	190	163	124	1	1	22	22	1	1	22	22	119
253	106	368	368	437	187	161	122	276	187	161	122	1	1	22	22	1	1	22	22	57
75	276	203	203	176	321	89	67	87	321	89	67	1	1	22	22	1	1	22	22	94
75	276	203	203	176	321	89	67	87	321	89	67	1	1	22	22	1	1	22	22	70

Процесс определения сонливости

После выделения лица и глаз на объекте выполняется измерение доли времени, в течение которого веки частично закрыты по оценкам соответствующей прикладной программы. Критерием сонливости является состояние, когда глаза закрыты больше 80% за минуту. Если процент закрывания глаз превышает 80% в течение минуты, то человек считается дремлющим.

Заключение

В статье рассмотрены вопросы построения подсистемы мониторинга и контроля усталости водителя. Проведен сравнительный анализ существующих категорий методов решения поставленной задачи, выявлены достоинства и недостатки. Определены направления в решении подзадачи распознавания и трекинга. В процессе исследования предметной области были выявлены существенные проблемы, которые могут возникнуть при обработке фоторяда, а именно сложность распознавания лица при изменении угла съемки и изменении освещения, а также проблемы в мобильных сервисах при использовании нескольких камер смартфона одновременно.

Литература

1. Culjak I., Abram D., Pribanic T., Dzapo H., Cifrek M. A brief introduction // 2012 Proceedings of the 35th International Convention MIPRO. Opatija, Croatia. 2012. P. 1725-1730.
2. Kalman R.E. A new approach to linear filtering and prediction problems // Journal of Basic Engineering, Vol. 82(1), 1960. P. 35-45.
3. Cristianini N., Shawe-Taylor J. An Introduction to Support Vector Machines and Other Kernel based Learning Methods. New York: Cambridge University Press, 2000.
4. Dalal N., Triggs B. Histograms of oriented gradients for human detection // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2005. Vol. 1. P. 886-893.
5. Amos B., Ludwiczuk B., Satyanarayanan M. Openface: A general-purpose face recognition library with mobile applications, CMU School of Computer Science, Pittsburgh, Pennsylvania, United States, CMU-CS-16-118, 2016.
6. Viola P., Jones M. Rapid Object Detection using a Boosted Cascade of Simple Features // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Kaii, Hawaii, USA. 2001. Vol. 1. P. 511-518.
7. Bay H., Ess A., Tuytelaars T., Van Gool L. Speeded-Up Robust Features (SURF) // Computer Vision and Image Understanding. June 2008. Vol. 110(3). P. 346-359.
8. Лашков И.Б., Смирнов А.В., Кашевник А.М. Исследование и разработка подхода к построению интеллектуального мобильного сервиса для автоматизированной поддержки водителя транспортного средства // Научно-технический вестник информационных технологий, механики и оптики. 2015. Т. 15. № 6. С. 1130–1138.
9. Буй Тхи Тху Чанг, Фан Нгок Хоанг, В.Г. Спицын Распознавание лиц на основе применения метода Виолы–Джонса, вейвлет-преобразования и метода главных компонент // Известия Томского политехнического университета. 2012. Т. 320. № 5

УДК 004.91

WEB-ОРИЕНТИРОВАННАЯ СИСТЕМА ПЛАНИРОВАНИЯ И УЧЕТА НАУЧНО-ТЕХНИЧЕСКИХ МЕРОПРИЯТИЙ В ДОННТУ

Чайка В.А., Землянская С.Ю.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: valera_chaika@mail.ru

Аннотация:

Чайка В.А., Землянская С.Ю. Web-ориентированная система планирования и учета научно-технических мероприятий в ДонНТУ. Рассмотрено составление годового плана университета и его сопровождение. Предложена web-ориентированная система, которая автоматизирует большую часть работы. Представлена информационная модель и описаны необходимые технологии для реализации данной системы.

Annotation:

Chaika V.A., Zemlyansky S.Y. Web-based planning system for accounting scientific and technical activities in DonNTU. Drawing up the annual plan of the university and its maintenance is considered. The web focused system which automates the most part of work is offered. Information model is presented and necessary technologies for realization of this system are described

Общая постановка проблемы

В современном мире активно протекают процессы автоматизации. Они обусловлены тем, что человек в процессе своей трудовой деятельности может допустить ошибки, которые с легкостью могла бы избежать система. Переход от бумажной волокиты к электронной документации – очередной этап эволюции человечества. Автоматизация помогает в разнообразных сферах деятельности человека. Также она будет полезна в сфере науки.

Кратко о НИЧ

Научно-исследовательская часть (НИЧ) является подразделением университета, организующим проведение фундаментальных, поисковых и прикладных научно-исследовательских и опытно-конструкторских работ, а также иных видов работ, связанных с осуществлением научной деятельности университета.

Ключевыми процессами НИЧ являются:

- планирование, организация и учет научно-технических мероприятий (НТМ);
- подготовка и представление заинтересованным сторонам информационно-аналитических материалов;
- управление научно-исследовательской и инновационной деятельностью университета.

В рамках основных видов деятельности сотрудники НИЧ решают следующие задачи:

- организация, планирование и контроль научно-исследовательской и инновационной деятельности университета;
- проведение фундаментальных, прикладных исследований и опытно-конструкторских работ;
- обеспечение выполнения научно-исследовательских работ в соответствии с направлениями, координируемыми МОН ДНР;
- подготовка нормативных и распорядительных документов, инструктивных материалов, регламентирующих процессы и процедуры научной деятельности университета.

Из всего этого разнообразия задач рассмотрим задачу планирования и учета научно-технических мероприятий.

Текущая ситуация в НИЧ

Для проведения научно-технических мероприятий составляется годовой план университета. Его составление протекает в несколько этапов:

- подача заявок
- составление плана факультета
- составление плана университета

После утверждения годового плана необходимо заниматься отслеживанием подачи всех необходимых материалов по каждому из мероприятий, а также составлять отчетность.

Помимо этого не исключены ситуации, когда утвержденный годовой план университета следует отредактировать (отменить/изменить существующие мероприятия, добавить новое).

На текущий момент все эти задачи являются трудоемкими, и в них легко допустить ошибку. Весь процесс составления плана университета по проведению НТМ и жизненный цикл мероприятия, начиная от заявки и заканчивая подведением результатов в виде отчетов и сборников материалов, ложится на плечи сотрудников НИЧ.

Описание подсистемы

Для того чтобы избежать множества ошибок, сократить время, затрачиваемое на выполнение рутинной работы, и улучшить контроль за процессом организации и проведения НТМ, целесообразно ввести web-ориентированную подсистему планирования и учета научно-технических мероприятий в ДонНТУ.

Перечень улучшений, которые предлагаются данной системой:

- Удобная форма для ввода заявок на проведение мероприятия. Для того, чтобы избежать ошибок ввода, и чтобы сохранялся единый стиль информации предлагается форма ввода заявки, в которой большинство полей для заполнения будут представлять из себя выпадающий список, а пользователь будет выбирать необходимое для него значение.

- Уведомления о наступлении срока подачи материалов. Каждое мероприятие сопровождается материалами (пресс-релиз, пост-релиз, сборник, отчет). Ответственный секретарь оргкомитета обязан следить за тем, чтобы каждый материал был подан своевременно. Для того чтобы избежать отступления от контрольных сроков, система будет отправлять письмо на электронную почту с напоминанием.

- Упрощенное составление отчетности и приказов. Каждый отчет имеет определенный шаблон и в данном шаблоне большинство полей можно заполнить автоматически, оставив на внесение сотруднику лишь некоторые поля.

- Отслеживание подачи планов НТМ факультетов. Сотрудник НИЧ сможет в режиме реального времени увидеть, какие факультеты уже предоставили свои планы, а какие – нет.

- Формирование плана НТМ факультета “одной кнопкой”, аналогично плану университета, который формируется на основании всех планов факультетов.

- Указание системой на несоответствия в составлении графика НТМ на стадии ввода данных (если уже на этот день занято помещение, указана запрещенная тема и т.д.).

Рассмотрим контекстную диаграмму подсистемы, представленную на рис. 1 и опишем основную информацию, представленную на данной диаграмме.

Для работы системы необходима следующая информация:

1. Заявки от кафедр – перечень мероприятий, которые планирует провести кафедра;
2. Материалы – различного рода информация, связанная с конкретным НТМ (пресс-релиз, пост-релиз, статьи).

В результате обработки входных данных система формирует следующие документы:

1. Годовой план НТМ – утвержденный план проведения НТМ на последующий год, который охватывает все факультеты;

2. Отчеты – раз в полгода университет предоставляет в МОН отчет о проведенных мероприятиях, также раз в год НИЧ предоставляет ректору годовой отчет. Помимо этого, если НТМ проходит в республиканском или международном формате, то по результатам проведения такого НТМ также составляется отчет.

При организации проведения НТМ необходимо руководствоваться такими документами:

1. Положение об НТМ – документ, который регулирует порядок проведения НТМ, сроки и ограничения;

2. Действующее законодательство – может ограничивать темы, которые будут рассматриваться в НТМ.

Автоматизированная система разрабатывается для использования следующими лицами:

1. Ответственный секретарь оргкомитета. Отвечает за организацию и проведение своего мероприятия. На него возлагается ответственность за предоставление материалов.

2. Заместитель декана по науке. Составляет план факультета и вносит изменения в него в случае необходимости.

3. Специалист НИЧ. Сотрудник НИЧ, который занимается обработкой планов факультетов, составляет полугодовые и годовые отчеты.

4. Начальник НИЧ. Регулирует сроки подачи документации, утверждает годовой план университета.

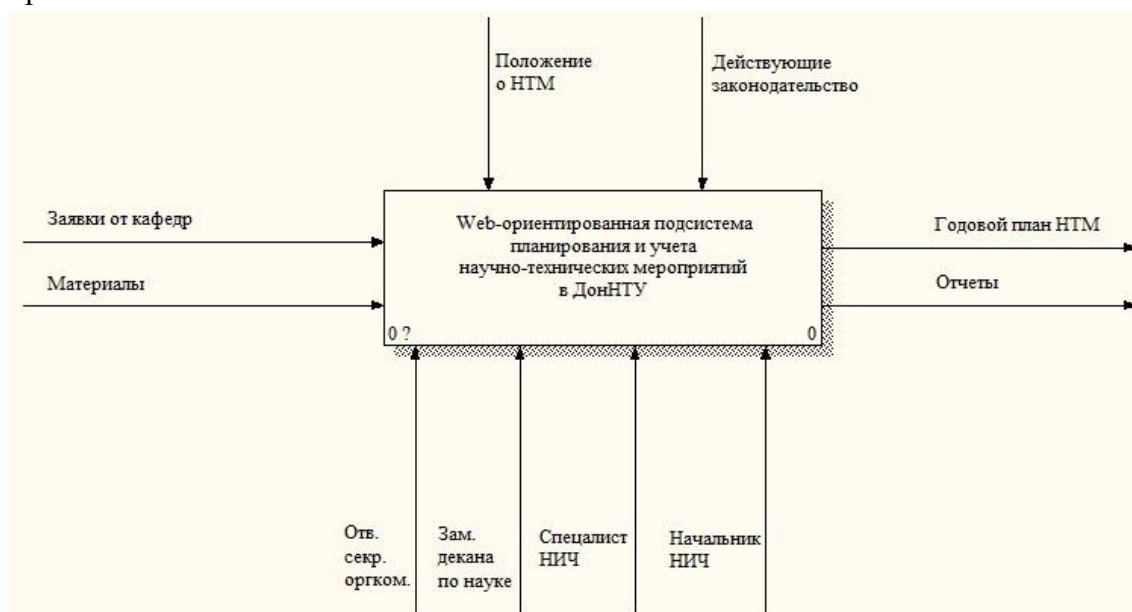


Рисунок 1 – Контекстная диаграмма подсистемы

Для наглядного отображения основных функций системы воспользуемся декомпозицией первого уровня, которая представлена на рис. 2

Рассмотрим каждую функцию подробнее.

1. Обработка заявок

Данная функция включает первичное исправление заявок, внесение их в электронную базу данных, а также проверку на наличие “наложений” (например, если на указанную дату уже занята аудитория, которая указана в заявке). Также имеется возможность отправки письма ответственным секретарям о том, что их заявка рассмотрена и внесена в базу данных.

2. Составление годового плана

Прежде всего, каждый факультет составляет свой собственный план - "план факультета". Этот план заместитель декана соответствующего факультета отправляет на рассмотрение в НИЧ, и если специалист НИЧ не видит никаких ошибок, то он вносит данный план факультета как часть годового плана. После того, как все факультеты предоставили свои планы, специалист НИЧ составляет приказ на утверждение плана. Этот приказ подписывают все необходимые лица, и он возвращается специалисту НИЧ. Далее начальник НИЧ окончательно утверждает годовой план.

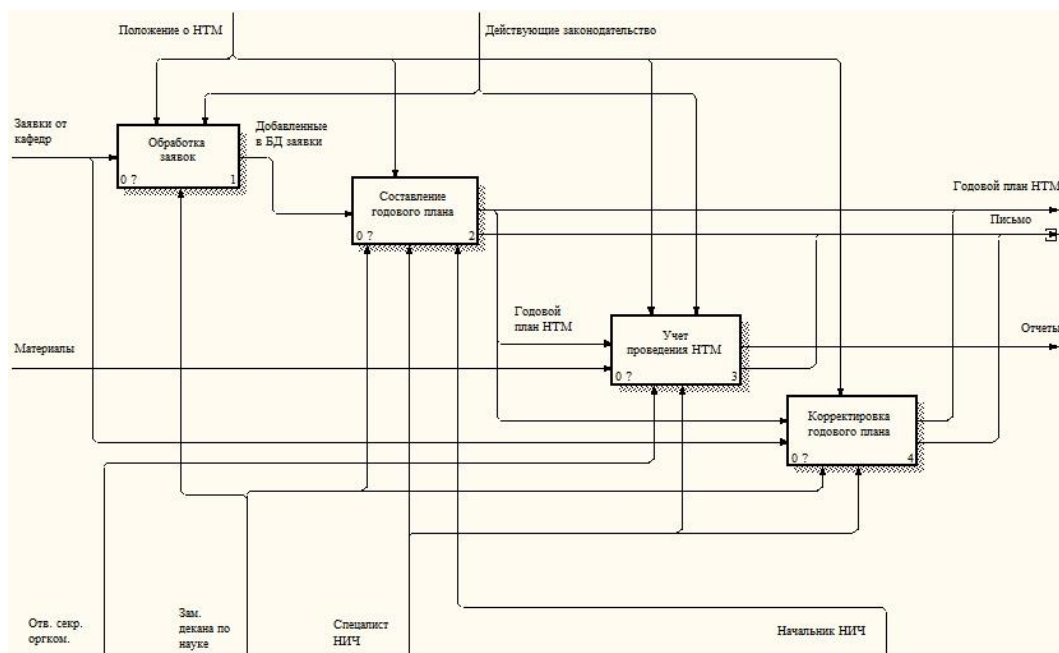


Рисунок 2 – Диаграмма декомпозиции первого уровня

3. Учет проведения НТМ

Данная функция подразумевает своевременное внесение материалов по каждому мероприятию и составление отчетности. Система отправляет электронные письма ответственным секретарям с напоминанием о необходимости добавления материалов.

4. Корректировка годового плана

После того, как годовой план утвержден начальником НИЧ, любая его корректировка производится исключительно после того, как заместитель декана напишет служебную записку на имя проректора по науке, а проректор по науке ее подпишет. Эта записка попадает в НИЧ и специалист НИЧ разрешает соответствующему факультету сделать необходимые изменения.

Информационная модель

Информационная модель данной подсистемы представлена на рисунке 3

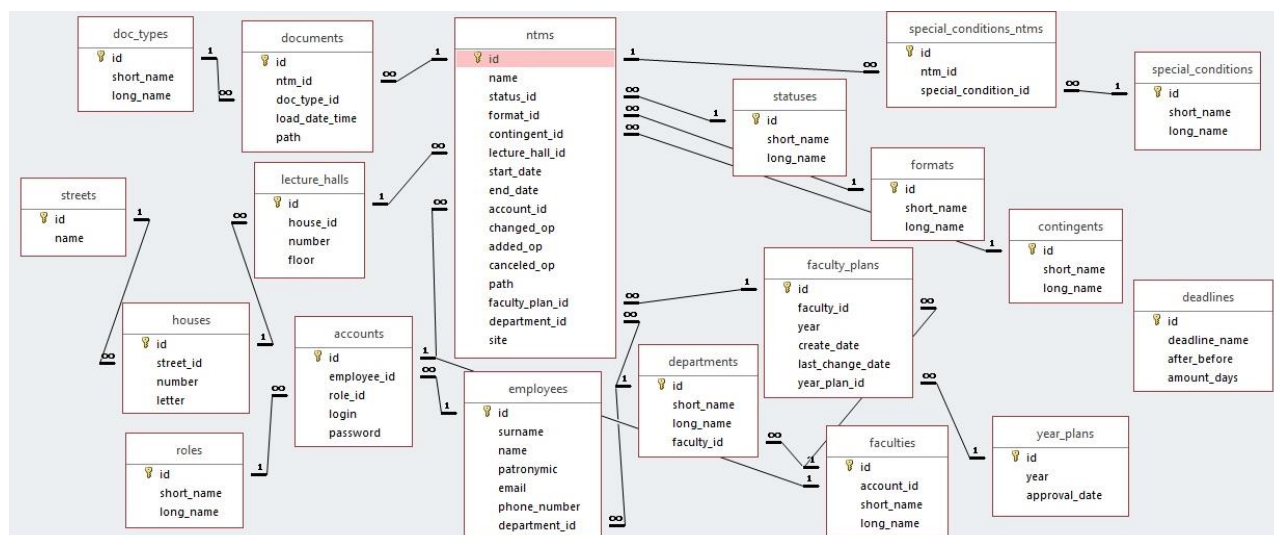


Рисунок 3 – информационная модель

Основными сущностями данной системы являются: НТМ, Документ, Сотрудник, План. Ключевой сущностью является “НТМ”. Она содержит поля, которые в полной мере описывают НТМ для данной подсистемы, например, можем определить тип мероприятия, даты проведения, ответственного секретаря, сайт и многое другое.

Технологии разработки ПО и его структура

Данная система является web-ориентированной в связи с рядом преимуществ систем такого типа:

- не требует мощного персонального компьютера;
- не требует установки;
- легкая доступность (при помощи мобильных устройств);
- разработка не требует покупки специальных программ.

Из недостатков данного типа систем можно выделить:

- неизбежны затраты на покупку и поддержания хостинга;
- для работы в системе необходимо подключение к интернету.

Технологии, которые будут использоваться для создания данной подсистемы:

HTML5, CSS3, Bootstrap 4, JQuery 3.4.0, PHP7.1, Laravel 5.8, MySQL.

Рассмотрим их предназначение.

Для разработки интерфейса данной подсистемы использовались HTML5, CSS3, Bootstrap 4. Благодаря HTML формируется “каркас” страниц, CSS позволяет стилизовать данный каркас, а Bootstrap позволяет в упрощенной форме адаптировать интерфейс под различный тип устройств, начиная от мобильных телефонов и заканчивая широкоформатными мониторами компьютеров.

Для того, чтобы реализовать всплывающие окна и анимировать некоторые элементы дизайна, применяется библиотека языка JavaScript – jQuery.

Серверная часть подсистемы написана на языке PHP, данный язык является основным при разработке web-ориентированных систем. Однако на «чистом» PHP реализация некоторых задач подсистемы весьма трудоёмка, поэтому было решено использовать фреймворк Laravel. Проекты, разрабатываемые с применением данного фреймворка, используют модель MVC, что является негласным стандартом в современной web-разработке.

Электронная база данных реализована с применением СУБД MySQL.

Выводы

Был проанализирован процесс составления и сопровождения годового плана проведения НТМ. Исходя из этого, выделены основные функции, спроектирована

информационная модель, предложена и описана web-ориентированная система для упрощения организации указанного процесса, выбраны технологии, с помощью которых она может быть реализована. Внедрение предлагаемой системы позволит сократить временные затраты, уменьшить количество ошибок и упростить общий контроль за всем жизненным циклом НТМ.

Литература

10. НИЧ ДонНТУ Положение об организации, проведении и учете научных и научно-технических мероприятий, проводимых в Донецком национальном техническом университете [Текст], - 2018 – С. 9-12.

11. Научно-исследовательская часть / Интернет-ресурс – Режим доступа : [www/URL: http://rgups.ru](http://rgups.ru) – Главная-Наука-Научно-исследовательская часть.

12. Laravel по-русски / Интернет-ресурс – Режим доступа : [www/URL: https://laravel.ru](http://www.laravel.ru) – Документация.

УДК 004.046

ВЕБ-ОРИЕНТИРОВАННАЯ СИСТЕМА МОНИТОРИНГА СОСТОЯНИЙ ЗАКАЗОВ ПРИ ПРИОБРЕТЕНИИ ТОВАРОВ В ИНТЕРНЕТ-МАГАЗИНЕ

Калюга Д.В., Поляков А.И., Теплова О.В

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: dannypopkorn@gmail.com

Аннотация:

Калюга Д.В., Поляков А.И., Теплова О.В. Веб-ориентированная система мониторинга состояний заказов при приобретении товаров в интернет-магазине. В статье рассмотрено продолжение развития электронной коммерции с углубленным использованием web-технологий для повышения эффективности систем online продаж товаров населению.

Annotation:

Kalyuga D.V., Polyakov A.I., Teplova O.V. Web-based system for monitoring the status of orders when purchasing goods in the online store. The article considers the continuation of e-Commerce development with in-depth use of web-technologies to increase online sales systems efficiency.

Общая постановка проблемы

В современных условиях развития товарно-денежных отношений бурное развитие получило такое явление как электронная коммерция.

Электронная коммерция - предпринимательская деятельность по продаже товаров, выполнению работ и оказанию услуг, осуществляемая с использованием электронных сообщений [1].

Онлайн торговля занимает достаточно большую нишу в торговле вообще, причем эта ниша постоянно увеличивается. Уже сейчас можно не выходя из дома купить любые вещи — достаточно лишь найти соответствующий магазин и иметь достаточно денег на банковском или электронном счете [2].

Исследования

Объектами исследования в данной статье являются информационные процессы, происходящие при совершении лицами покупок через интернет-магазин. Интернет-магазин — это совокупность программ, работающих на online-ресурсе, которые позволяют покупателю дистанционно выбрать товар из каталога и оформить его заказ [3].

Проанализировав рынок, можно сделать вывод, что для повышения рейтинга среди подобных систем, необходимо использовать эргономический удобный расширенный функционал, в котором пользователю предоставляется возможность не только просмотреть данные о своём заказе, но и отслеживать состояние заказа в online-режиме.

Разрабатываемая система должна помимо возможности заказать товары, предоставляет еще и возможность отслеживать заказы без сторонних сервисов. При этом пользователь будет экономить трафик и собственное время, так как не будет искать состояния своих заказов на сервисах сторонних организаций, а сможет просмотреть все данные заказа в единой системе.

Цели данной информационной системы разделяются на 2 категории:

1. С точки зрения клиента
2. С точки зрения системы

С точки зрения клиента целью системы является повышение качества обслуживания клиентов за счет удобного интерфейса.

С точки зрения системы целями являются:

- 1) Снижение временных затрат на обработку заявок от клиентов.
- 2) Снижение количества ошибок при оформлении документации.

В настоящее время можно выделить следующие средства автоматизации типового магазина:

- 1) Коммерческая система 1С-Битрикс.

1С-Битрикс – это программный продукт для управления содержимым сайта (CMS) [4].

- 2) Система Wordpress.

WordPress — это система управления содержимым сайта (CMS) с открытым исходным кодом, распространяемая под лицензией GNU GPL версии 2. Написана на PHP, в качестве базы данных использует MySQL [5].

Перечисленные системы не удовлетворяют всем потребностям, необходимым для разработки такого рода систем, так как они являются коммерческими и закрытыми, то есть в них нет допуска к исходным кодам.

В результате проведенного анализа объекта исследования были определены основные параметры информационно-материального потока, действующие лица и накладываемые на автоматизированную информационную систему ограничения. Полученная контекстная диаграмма веб-ориентированной системы мониторинга состояний заказов при приобретении товаров в интернет-магазине реализована в СА ERWin Process Modeler в нотации IDEF0 и представлена на рис. 1.

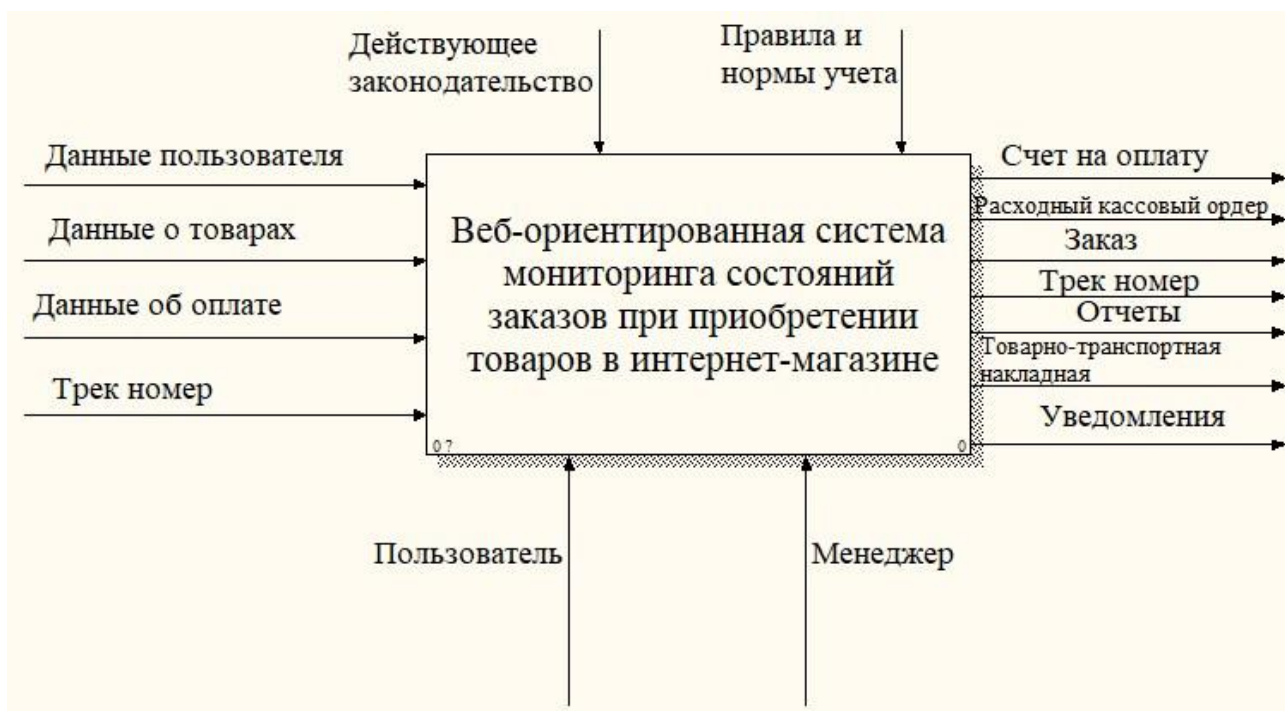


Рисунок 1 – Контекстная диаграмма разрабатываемой системы

Данная система основана на web-ориентированном подходе, что позволяет пользователям получать к ней доступ с помощью любого устройства. Такая система очень удобна для клиентов интернет-магазина, так как она позволяет просматривать каталог товаров, выбирать и оформлять заказы без личного присутствия. Так же она позволяет просматривать данные о оформленном заказе по трек-номеру.

Трек-номер - это уникальный цифровой или буквенно-цифровой идентификатор, который присваивается почтовому отправлению [6].

Информационная модель системы будет иметь следующий вид (рис. 2)

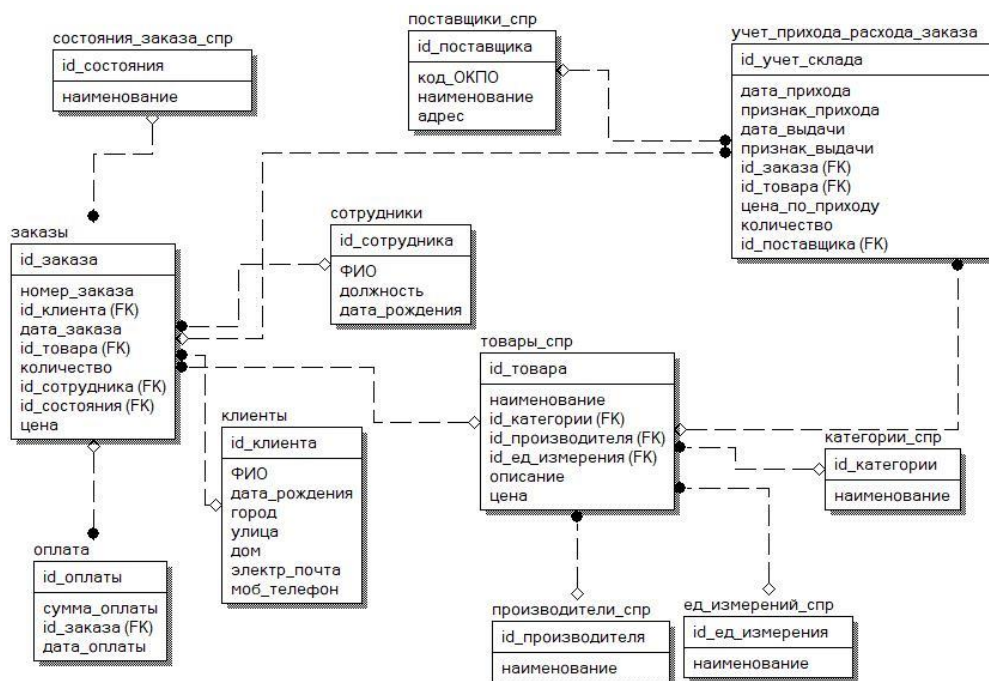


Рисунок 2 – Информационная модель разрабатываемой системы

Разрабатываемая система должна выполнять следующие функции:

1. Регистрация и авторизация пользователей в системе.
2. Работа с заказом. В системе имеется возможность добавления товаров в корзину из списка, сформированного с использованием различных фильтров, интересующих клиента. После того как заказ обработан и подтверждён, система формирует конечный заказ, трек-номер и счет на оплату заказа. По трек-номеру пользователь может просматривать различную информацию о заказе.

3. Обработка данных об оплате и выдача товаров. Если пользователь оплатил заказ верно, то система сформирует товарно-транспортную накладную, изменит состояние товара на Оплачено и уведомит клиента о том, что платеж пришел. В случае того, что заказ пришел на склад, изменится состояние заказа и система уведомит покупателя об этом. Если оплата была произведена неверно, то система вернет деньги на тот счет, с которого была произведена оплата.

4. Формирование различных видов отчетов по задаваемым параметрам, например, за период, по продавцу, покупателю, по товарам, и т.д.

После определения внешних информационных потоков на контекстной диаграмме, можем перейти к декомпозиции системы (рис. 3).

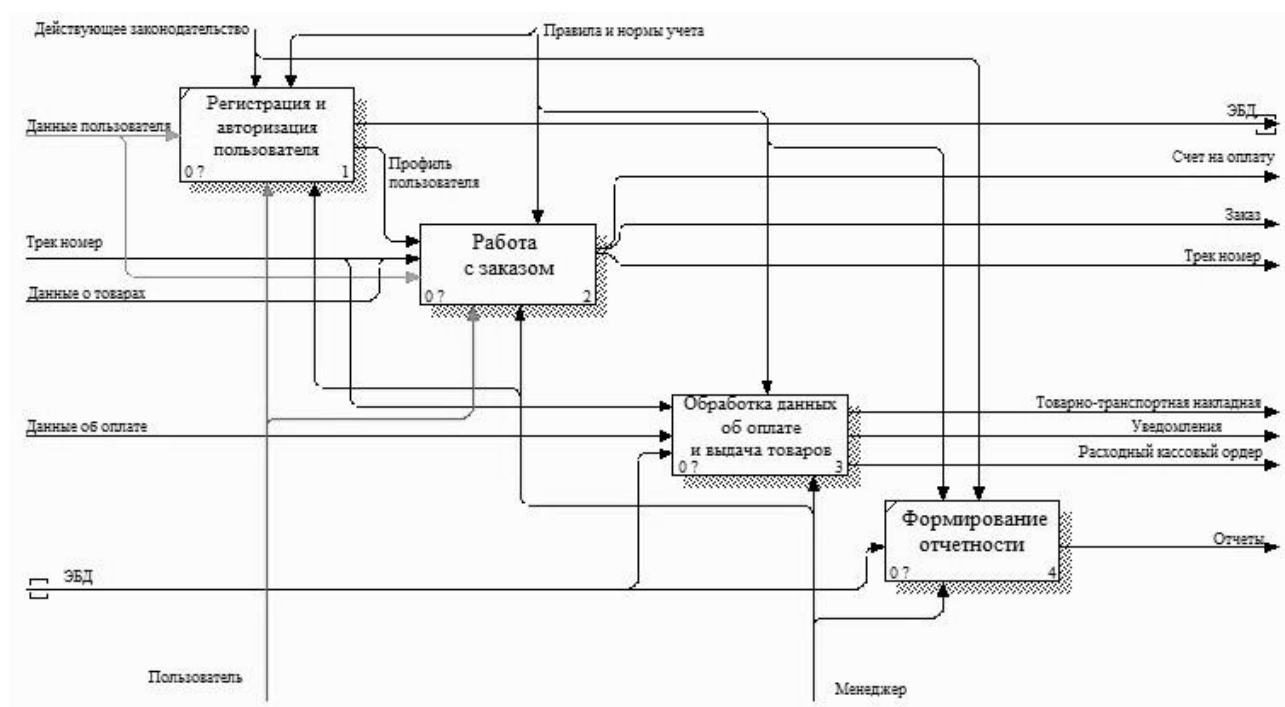


Рисунок 3 – Декомпозиция первого уровня разрабатываемой системы

Выводы

Таким образом, внедрение системы мониторинга состояний заказов при приобретении товаров в интернет-магазине повысит эффективность деятельности и уменьшит затрату временных ресурсов и интернет трафика для заказов товаров и отслеживания их.

Литература

1. Определение электронной коммерции [Электронный ресурс]. Режим доступа: <https://studfiles.net/preview/1476918/> (дата обращения: 01.04.2019).
2. Бирюков, П. И. Интернет-шопинг / П.И. Бирюков. - М.: Феникс, 2017. - 160 с.
3. Что такое Интернет-магазин? [Электронный ресурс]. – Режим доступа: <https://www.expertplus.ru/information/stati/22> (дата обращения: 01.04.2019).

4. Что такое 1С-Битрикс? [Электронный ресурс]. – Режим доступа: <https://siteblogger.ru/raznoe/chto-takoe-1s-bitriks.html> (дата обращения: 01.04.2019).

5. Что такое Wordpress (ВордПресс)? [Электронный ресурс]. – Режим доступа: <https://hostenko.com/wpcafe/tutorials/chto-takoe-wordpress/> (дата обращения: 01.04.2019).

6. Что такое трек-номер? [Электронный ресурс]. – Режим доступа: <https://shopper.life/page/faq/track> (дата обращения: 01.04.2019).

7. О'Коннор, Макдермотт И. Искусство системного мышления: необходимые знания о системах и творческом подходе к решению проблем. — М.: Альпина Бизнес Букс, 2006 — 256 с.

УДК 004.93'1

СОВРЕМЕННЫЕ ТЕНДЕНЦИИ В ТЕХНОЛОГИИ МАШИННОГО ОБУЧЕНИЯ

Егорова М.С., Мартыненко Т.В.

Донецкий национальный технический университет
кафедра автоматизированных систем управления
email: mariya.yegorova.mail@gmail.com

Аннотация:

Егорова М.С., Мартыненко Т.В. Современные тенденции в технологии машинного обучения. В статье рассмотрены основные новации в области методологии машинного обучения. Выполнен анализ современной научной литературы, посвященной вопросам развития методологии и областей прикладного использования рассматриваемых тем. Сформулированы предположения о будущих тенденциях и предложены наиболее перспективные направления исследований.

Annotation:

Yegorova M.S., Martynenko T.V. Contemporary trends in machine learning technology. In this article we reviewed the main innovations in the field of machine learning methodology. Also an analysis of modern scientific literature devoted to the development of methodology and areas of applied employment of the issues, we are considering, has been carried out. In addition, assumptions were formulated about future trends in the development of machine learning and suggested the most promising areas of research.

Постановка проблемы в общем виде

Все большее внимание исследователей занимает вопрос методологий или метамоделей (от англ. metamodel): принципов использования, комбинирования и выбора конкретных моделей и методов машинного обучения. Многолетний прогресс в разработке методов машинного обучения породил не только разнообразные решения, но и встретил на своем пути немало препятствий.

Основная трудность, с которой сталкивается человек в процессе знакомства с областью машинного обучения, – огромное количество разрозненных методов, каждый из которых обладает своими особенностями, областью использования и преимуществами. С развитием математических и алгоритмических методов становится все труднее хорошо ориентироваться во всех нюансах применяемых алгоритмов.

Цель статьи – обзор актуальных приемов, технологий и методик, применяемых при решении прикладных задач машинного обучения, по материалам научных статей в высокорейтинговых журналах зарубежных исследователей, аналитических и обзорных

заметок из открытых источников, а также технической документации и пресс-релизов технических и программных решений.

Проблема интерпретируемости

В качестве математической и инструментальной основы машинного обучения сегодня центральное место занимают искусственные нейронные сети. Они стали своеобразным универсальным языком представления обучаемых моделей. И хотя искусственный интеллект и машинное обучение как области знаний гораздо шире и включают целые семейства других методов, нейросетевые модели в настоящий момент так или иначе фигурируют в 90% научных публикаций в данных областях [1].

С разработкой все более сложных и глубоких архитектур нейронных сетей вместе с несомненными достоинствами все явственнее проявляются общие проблемы этого подхода. Наиболее существенной из них можно назвать принцип работы любой нейросети как «черного ящика» – даже должным образом обученная сеть не дает исследователям информации о внутренней структуре проблемы и выявленных зависимостях в данных. Обученная нейросеть является набором матриц весов, и смысловая интерпретация этих весов в общем случае не предполагается. С этой точки зрения нейросети представляют собой лишь инструмент решения конкретной задачи машинного обучения, но не дают экспертам аналитической информации для исследования проблемы. Этот недостаток заставляет исследователей заниматься проблемами интерпретируемости нейросетевых моделей [2, 3].

В настоящее время выделяют класс так называемых «естественно интерпретируемых моделей» – моделей, человеческая интерпретация которых не представляет труда в силу самой архитектуры модели. К ним относят, в частности: линейные и логистические регрессии, деревья решений, наивные байесовские классификаторы, k ближайших соседей, модели правил вывода и другие [3]. Объяснение внутреннего устройства других типов моделей, в том числе глубоких нейронных сетей, представляет собой открытую научную задачу. Основные методы для построения моделенезависимых объяснительных механизмов (англ. model-agnostic methods) представлены в таблице 1.

Таблица 1. Методы построения механизмов

Метод	Назначение
Функции и графики частичных зависимостей	Показывают предельное влияние признаков на результирующее значение
Механизм индивидуального условного ожидания	Расширение метода функций частичных зависимостей на локальный анализ отдельных предсказаний
Исследование уверенности модели (англ. model class reliance)	Моделенезависимый вариант алгоритма важности признаков, который позволяет оценить зависимость средней ошибки модели в зависимости от дисперсии отдельных признаков
Метод LIME (от англ. local interpretable model-agnostic explanations)	Основан на обучении локальных интерпретируемых моделей по предсказаниям модели, исследуемой для объяснения конкретных предсказаний.

В целом, область исследований интерпретируемости моделей машинного обучения можно назвать очень актуальной. Несомненно, данная проблематика получит свое развитие в дальнейших исследованиях в ближайшее время.

Предобученные модели

Использование нейронных сетей неразрывно связано с процессом их обучения. Даже самая продвинутая и сложная глубокая нейронная сеть, будучи необученной на правильно выбранном наборе данных, не может решить даже самых простых задач анализа данных.

Эффективность работы обученных нейросетей напрямую зависит от объема и качества обучающей выборки и времени обучения. А время обучения, в свою очередь, зависит от вариативности модели. Чем более вариативная, глубокая и сложная модель используется, тем больше времени на обучение она требует и тем больший объем данных нужен для предотвращения переобучения.

Таким образом, перед исследователями стоит не только задача поиска качественной обучающей выборки и очистки данных, но и выбора архитектуры сети и проведения довольно затратного в плане процессорного времени обучения сети. Это приводит к вполне естественному предположению о возможности повторного использования обученных нейросетей. Существует несколько способов использования предобученных моделей в зависимости от размера доступной исследователям выборки и схожести выборок, используемых для предобучения и непосредственно для анализа (таблица 2).

Таблица 2. Предобученные модели

Наименование	Описание
Автоматизированное создание признаков	Предобученные модели могут использоваться для автоматического извлечения значимых признаков из анализируемой выборки данных для последующего анализа
Использование только архитектуры	Предобученные модели, показавшие свою эффективность на определенном модельном наборе данных, могут послужить образцом архитектуры для построения новой модели и обучения ее на анализируемом наборе
Частичное дообучение	Предобученные модели используют частично, а обучение происходит для части слоев такой сети. Обычно веса начальных слоев замораживают, а конечных обучают на непосредственно анализируемой выборке
ANN-HMM (от англ. artificial neural networks – hidden Markov models)	Использование глубоких сетей доверия для начальной инициализации весов нейронной сети с последующим дообучением методом обратного распространения ошибки

Использование предобученных нейронных сетей, несомненно, является актуальным направлением развития технологии машинного обучения. Однако в настоящее время широкое распространение переноса обучения затруднено, в частности, отсутствием общепринятого формата хранения и распространения моделей нейронных сетей.

Мультизадачные сети

Одной из характеристик классических систем машинного обучения является специфичность модели для определенной задачи. Более того, различные архитектуры нейронных сетей имеют свою специфику в отношении предметной области. Естественным является вопрос о построении универсальной нейронной сети, которую можно было бы с минимальным переобучением использовать для решения многих различных задач.

В 2017 г. произошел прорыв в области многозадачного обучения. Команда Google Brain выпустила работу [4], в которой представила универсальную многокомпонентную архитектуру нейронной сети, названную MultiModel, которая достигает высоких результатов в одновременном обучении задачам из совершенно разных областей: обработка естественного языка, машинный перевод, распознавание лиц на изображении, распознавание речи. Важным методологическим результатом стало то, что такая сеть способна улучшать результаты одной задачи с обучением другой, совершенно, на первый взгляд, с ней не связанной.

Область мультизадачного обучения в ближайшие годы получит дальнейшее развитие. Представляется весьма интересным анализ мультизадачных сетей методами интерпретации черного ящика, развивающихся сегодня параллельно.

Нейроэволюция

Искусственные нейронные сети доказали свою эффективность в решении многих задач, однако эта эффективность в большой мере зависит от умения подбирать правильную архитектуру сети под каждую конкретную задачу анализа данных. При проектировании архитектуры сети эксперт сталкивается с необходимостью принять множество решений, как количественных, так и качественных, от реализации которых напрямую зависит производительность результирующей модели. В процессе такого проектирования задача эксперта – балансирование между вариативностью модели и склонностью к переобучению.

Пространство возможных нейронных сетей, в котором ведется поиск, огромно. При проектировании нейронных сетей специалисты используют определенные эвристические правила и инструменты диагностики, однако полноценной методологией такой поиск назвать сложно, это скорее творческий процесс.

Главным недостатком нейроэволюционного подхода являются весьма высокие требования к вычислительной мощности для поддержки данного процесса до сходимости. Развитие глубоких нейронных сетей и распространение больших данных еще выше поднимает эту планку. Поэтому, в настоящее время наблюдается спад интереса к эволюционному программированию искусственных нейронных сетей: время программиста и специалиста в области анализа данных сейчас стоит дешевле, нежели требуемое процессорное время.

Однако большие корпорации, располагающие высокими вычислительными мощностями, могут позволить себе эксперименты с нейроэволюцией. С ростом производительности вычислительной техники интерес к эволюционным методам в машинном обучении будет возвращаться (возможно итерационно), пусть не в ближайший год, но в обозримом будущем.

Автоматизированное обучение

На протяжении последнего времени автоматизация машинного обучения стала широко обсуждаемой темой и одной из наиболее быстрорастущих областей теоретических и практических разработок. Р. Олсон, один из разработчиков библиотеки автоматизированного машинного обучения, выделяет три главных процесса, автоматизация которых возможна и высвобождает большое количество временных ресурсов экспертов:

- подбор гиперпараметров моделей;
- испытание большого количества разных моделей;
- использование разных признаков, выделенных из данных [5].

Сложность традиционного подхода к построению систем машинного обучения состоит в необходимости знания всех существующих алгоритмов искусственного интеллекта, умения их правильно применить и настроить. Предлагаемые методы автоматизации машинного обучения связаны с существующими инструментальными средствами моделирования. Большинство из них ориентируется на библиотеку `scikit-learn` языка Python. Также существуют решения, основанные на генетическом подходе [6].

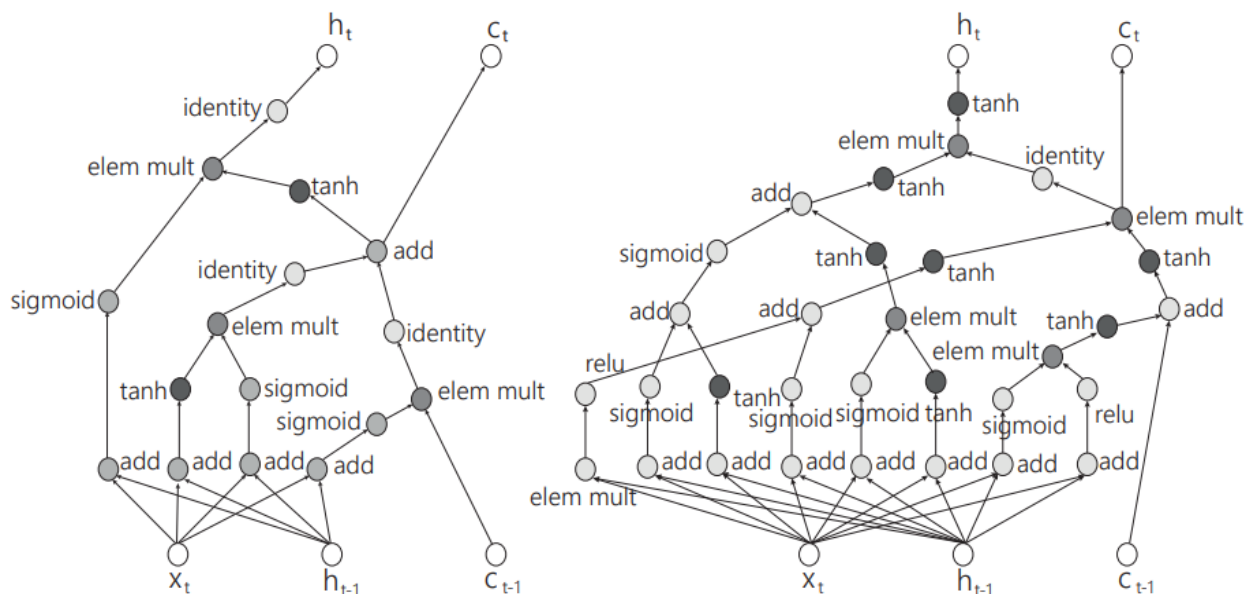


Рисунок 1 – Принципиальная схема одного элемента нейронной сети распознавания изображений, созданная человеком (слева) и с помощью Google AutoML (справа)

Активным направлением исследований в области автоматизации машинного обучения является проработка вопроса использования сложных нелинейных конвейеров обработки данных. Преимущественным методом нахождения таких конвейеров является генетическое программирование. В середине прошлого года исследовательский отдел Google представил архитектуру AutoML, основанную на обучении с подкреплением [7]. Эта система строит рекуррентные сети, схожие по своей архитектуре с построенными человеком, но более сложные. В конце 2017 г. усовершенствованная система AutoML смогла построить сеть, превосходящую все существующие в задачах распознавания объектов на изображении. Схема элемента представлена на рисунке 1.

Конечно, инструменты автоматизации машинного обучения не способны полностью вытеснить человека как участника процесса обработки данных. Также один из барьеров перед использованием систем AutoML – повышенные требования к вычислительным ресурсам. Однако с учетом экономии времени проектирования преимущество использования автоматизированного машинного обучения неоспоримо. Таким образом, можно с уверенностью назвать автоматизацию машинного обучения актуальной и активно развивающейся областью исследований. Поиск новых методов выбора моделей, перекрестной проверки, эволюционного и аналитического подбора алгоритмов обучения представляет как научный, так и чисто практический интерес.

Выводы

В настоящее время наблюдается прогресс в развитии методик автоматизированного поиска путей построения эффективных обучающихся моделей анализа данных, применимых ко многим практическим задачам интеллектуального анализа данных. В ходе обзора современных тенденций в машинном обучении нами выделены следующие перспективные направления фундаментальных и прикладных исследований в данной области:

1. Теоретические исследования в области интерпретируемости моделей искусственного интеллекта в сочетании с анализом автоматически построенных моделей.
2. Практические исследования мультизадачных, генеративных моделей.
3. Распространение автоматизированных средств машинного обучения.

4. Развитие и унификация инструментальных средств, в том числе облачных средств и сервисов интеллектуального анализа данных.

5. Разработка новых интеллектуальных продуктов пользовательского уровня, основанных на вышеперечисленных достижениях методологии машинного обучения.

Литература

1. Хохлова Д., «Бум нейросетей: Кто делает нейронные сети, зачем они нужны и сколько денег могут приносить». Режим доступа: <https://vc.ru/16843-neural-networks>.

2. Koh P.W., Liang P., “Understanding black-box predictions via influence functions” // Proceedings of the 34th International Conference on Machine Learning, PMLR. Vol. 70. P. 1885–1894.

3. Molnar C., “Interpretable machine learning”. Режим доступа: <https://christophm.github.io/interpretable-mlbook/>.

4. Kaiser L. [и др.], “One model to learn them all”. ArXiv:1706.05137 [cs.LG]. Режим доступа: <https://arxiv.org/abs/1706.05137>.

5. Olson R., “TPOT: A Python tool for automating data science”. Режим доступа: <https://www.kdnuggets.com/2016/05/tpot-python-automating-data-science.html/2>.

6. Olson R.S., Moore J.H., “TPOT: A tree-based pipeline optimization tool for automating machine learning”. P. 66–74.

7. Zoph B. & Le Q.V., “Neural architecture search with reinforcement learning”. Режим доступа: <https://arxiv.org/abs/1611.01578>.

УДК 004.046

АВТОМАТИЗИРОВАННАЯ СИСТЕМА УЧЕТА ЗАКАЗОВ КЛИЕНТОВ/ПОСЕТИТЕЛЕЙ РЕСТОРАНА

Головин П.В., Поляков А.И., Пряхин В.В.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: pasha.golovin.don@gmail.com

Аннотация:

Головин П.В., Поляков А.И., Пряхин В.В. Автоматизированная система учета заказов клиентов/посетителей ресторана. В данной статье, рассмотрена система, внедрение которой позволит значительно ускорить обслуживание клиентов в ресторане, сделать работу сотрудников более удобной, заинтересовать новых посетителей, увеличить доход предприятия и создать возможность рационального использования ресурсов.

Annotation:

Golovin P.V., Polyakov A.I., Pryakhin V.V. Automated system of accounting of customers' orders/restaurant visitors. In this article, the system is considered, the introduction of which will significantly speed up customer service in the restaurant, make the work of employees more convenient, interest new visitors, increase the company's income and create the possibility of rational use of resources.

Общая постановка проблемы

В настоящее время автоматизация производства проникает всё глубже и глубже во все сферы промышленности и сферы услуг, не является исключением и сфера общественного питания.

Последнее время, все больший интерес к автоматизации явно ощущается со стороны активно развивающихся предприятий общественного питания, владеющих достаточным количеством средств, желающих минимизировать потери рабочего времени персонала, сэкономить на его численности и поднять уровень сервиса в заведениях.

Исследования

На рынке существует несколько продуктов для комплексной автоматизации ресторанов, такие как: R-Кеерер, iiko.

R-Кеерер - торговая марка компании UCS, под которой продаются программное обеспечение и программно-аппаратные комплексы. Разработана в Москве в 1992 году, т.е. R-Кеерер - это 20-летний опыт работы на рынке автоматизации ресторанов.

Разрабатывается российской компанией UCS Россия, г. Москва. с 1992 года. На данный момент (июль 2016) актуальной является версия R-Кеерер v7. Широко распространена предыдущая версия системы - R-Кеерер v6. В 2007 году R-Кеерер занимал 34% рынка автоматизации ресторанов, автоматизировав более 9500 ресторанов. В апреле 2009 года было автоматизировано 15000 ресторанов. По данным разработчика на лето 2011 года, R-Кеерер установлен более чем в 20000 ресторанах, кафе, ресторанах типа FastFood, клубах и развлекательных центрах в 30 странах мира, а по данным на июль 2016 года R-Кеерер установлен уже в 42000 ресторанах в 47 странах мира [5].

iiko - проект, созданный сооснователем АБВУУ Давидом Яном, занимает второе место по популярности в России. Система характеризуется наличием большого числа различных модулей, которые подключаются к специальному серверу и работают через него. Есть функциональность фронт- и бэк-офиса.

Проект создания системы iiko был запущен в августе 2005 г. Давидом Яном и Максимом Нальским, в начале 2008 г. компания «Тройка Диалог» инвестировала несколько миллионов долларов в развитие проекта. Большая часть команды - разработчики ПО

2006 год — Прототип iikoRMS начал работать в FAQ-cafe

2007 год — На выставке ПИР в Москве презентована первая версия iikoRMS

2008 год — С компанией «Планета гостеприимства» подписан контракт, согласно которому компания iiko автоматизирует все рестораны «Сбарро», «Восточный Базар», «Виаджио» и «Баш-на-Баш» в России (более 130 предприятий)

2009 год - iiko выходит на второе место по объему продаж на рынке автоматизации ресторанного бизнеса

2010 год - об успешном окончании пилотного проекта и поэтапном переходе на iiko заявляет сеть Кофе Хауз (более 200 кофеен «Кофе Хауз», а также проекты «Азия Кафе» и «Винегрет Кафе») [6].

При анализе данных систем, можно выделить следующие функции:

1. Работа с текущими заказами клиента
2. Бронирование столиков
3. Бронирование мероприятий
4. Распределение персонала
5. Формирование отчетности

Изучив вышеуказанные системы, можно произвести сравнительный анализ.

Результаты приведены в таблице:

Таблица 1 – Сравнение существующих систем с разрабатываемой

Функция \ Система	R-keeper	iiko	АС учета заказов клиентов/посетителей ресторана
Бронирование столика	+	+	+
Бронирование мероприятия	-	+	+
Работа с текущими заказами без присутствия официанта	-	+	+
Взаимодействие с кухней	+	+	+
Распределение персонала	-	-	+
Отчетность	+	+	+
Стоимость	Большая	Большая	Средняя

Проанализировав результаты исследований, можно выделить:

Основные цели системы:

1. Упрощение оформления текущего заказа клиента
2. Ускорение и упрощение бронирования столиков/мероприятий.
3. Упрощение работы с персоналом

Назначение системы:

1. Автоматизация обслуживания заказов клиентов в ресторанном зале.
2. Автоматизация управления персоналом.

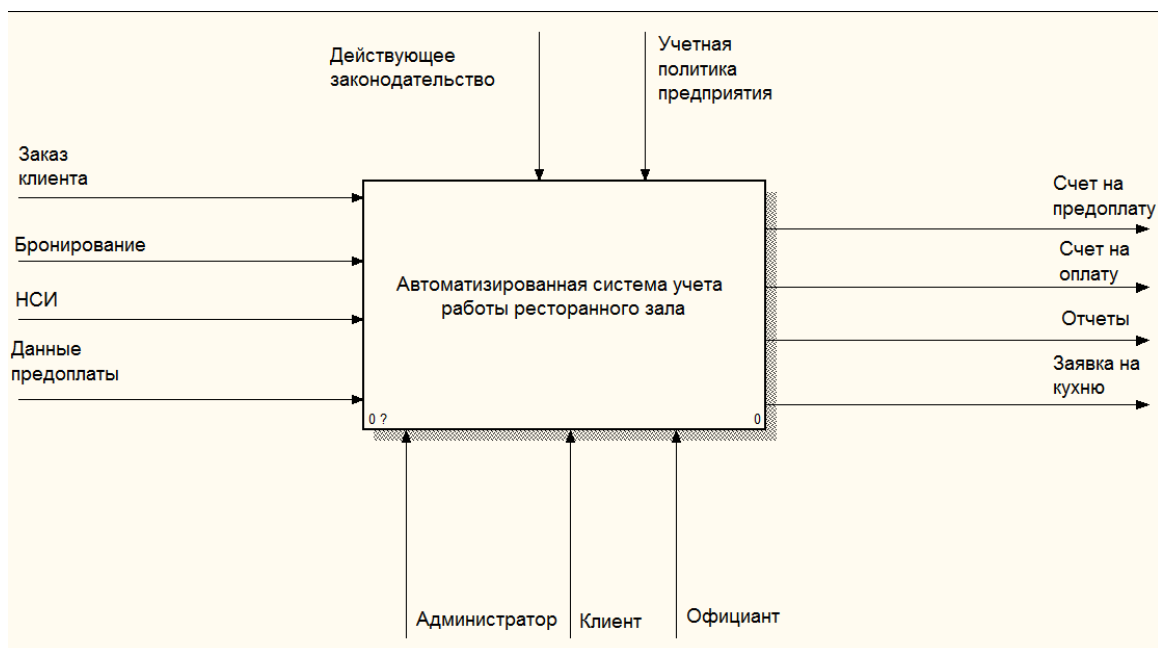


Рисунок 1 – Контекстная диаграмма разрабатываемой системы

Так как система работает по типу клиент-сервер, это значительно облегчает и оптимизирует работу официантов. Им не обязательно подходить к новому клиенту, официант будет приносить заказ, и выполнять какие-то пожелания клиента (будет реализована функция вызов официанта). Администратору будет удобно составлять график работы сотрудников, следить за мероприятиями и их организацией, формировать отчетность.

В ходе разработки системы, были определены следующие функции:

1. Бронирование столика
2. Бронирование мероприятия
3. Оформление заказа
4. Закрепление персонала за столиками
5. Формирование отчетности

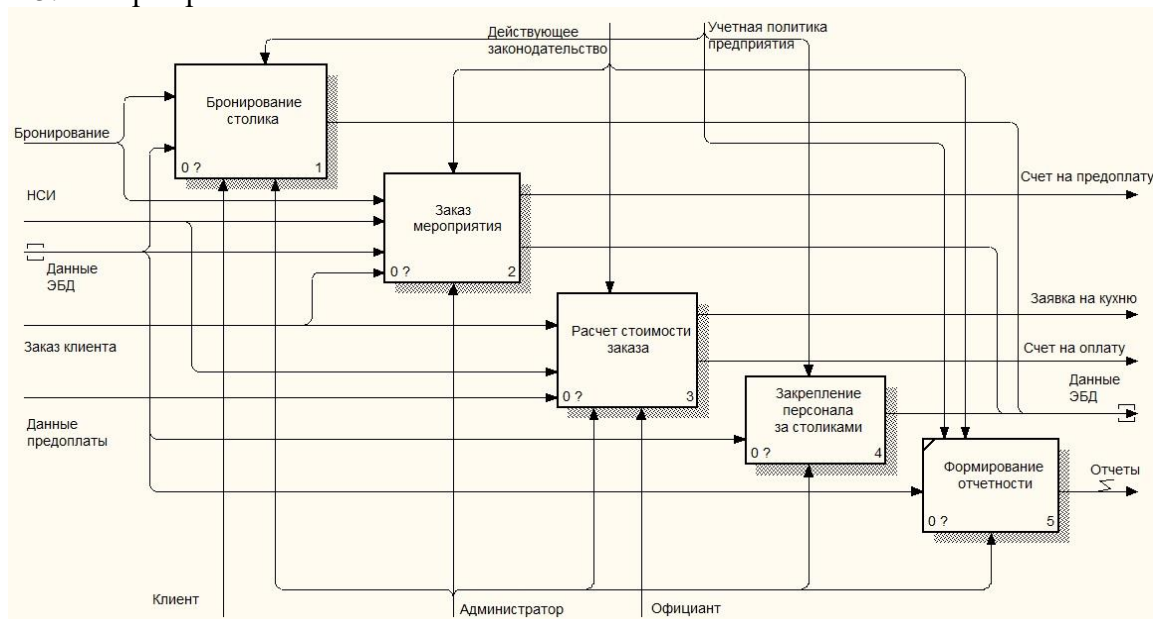


Рисунок 2 – Декомпозиция первого уровня разрабатываемой системы

При бронировании столика, существует 2 вида бронирования: онлайн, когда клиент с помощью приложения или сайта хочет забронировать столик, при этом система проверяет возможность брони учитывая занятость зала. При офлайн бронировании, клиент взаимодействует с администратором.

При бронировании мероприятия составляется его план: составление меню, исходя из которого формируется предоплата, назначение персонала, который будет присутствовать на мероприятии.

Оформление заказа подразумевает как обычный текущий заказ, так и мероприятие. При обычном, заказ клиента поступает в систему, затем официант подтверждает его, и передает заявку на кухню. Так же возможен учет дополнительных услуг на основе НСИ. В конце формируется окончательная стоимость как для обычного заказа, так и для мероприятия (необходимо учесть предоплату).

Работа с персоналом – это составление графика работы сотрудников, закрепление официантов за столиками, формирование заработной платы.

Выводы

Таким образом, внедрение данной системы, позволит значительно ускорить обслуживание клиентов в ресторане, сделать работу сотрудников более удобной, заинтересовать новых посетителей, увеличить доход предприятия и создать возможность рационального использования ресурсов.

Литература

1. Автоматизированные системы управления рестораном [Электронный ресурс]. Режим доступа: https://studref.com/306386/turizm/avtomatizirovannye_sistemy_upravleniya_restoranom (дата обращения: 01.04.2019).
2. Функции автоматизированной системы управления рестораном [Электронный ресурс]. Режим доступа: <http://prohotelia.com/2009/06> (дата обращения: 01.04.2019).
3. Значение и возможности АСУ в ресторанном бизнесе [Электронный ресурс]. Режим доступа: http://www.mnogosmenka.ru/bistrov/avtomatizirovannye_sistemy.htm (дата обращения: 01.04.2019).
4. Виды автоматизированных систем управления рестораном [Электронный ресурс]. Режим доступа: https://knowledge.allbest.ru/programming/3c0b65625b3bc78a4c43a89421206c36_0.html (дата обращения: 01.04.2019).
5. R-keeper [Электронный ресурс]. Режим доступа: https://ucs.ru/products/r_keeper (дата обращения: 01.04.2019).
6. Iiko [Электронный ресурс]. Режим доступа: <https://iiko.ru> (дата обращения: 01.04.2019).

УДК: 004.42

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ STEPTRACKER (ШАГОМЕР) В MIT APP INVENTOR

Харитонов Ю.Е., Денисенко Е.А.

ГОУВПО «Донецкий национальный технический университет»
кафедра экономической кибернетики

E-mail: Denisenko7737@yandex.ru

Аннотация:

Харитонов Ю.Е., Денисенко Е.А. Разработка мобильного приложения StepsTracker (шагомер) в MIT App Inventor. Рассмотрены теоретические основы облачной среды визуальной разработки приложений MIT App Inventor. Обобщено функциональное назначение разработанного приложения, представлено описание его работы и приведен пример расчета основных показателей.

Annotation:

Kharitonov Y.E., Denysenko Y.A. Development of the mobile application StepsTracker (pedometer) in MIT App Inventor. The theoretical foundations of the cloud environment for visual application development MIT App Inventor are considered. The functional purpose of the developed application is summarized, a description of its work is presented, and an example of the calculation of key indicators is given.

Общая постановка проблемы

С развитием технологий создания персональных гаджетов, рынок мобильных приложений получил мощнейший стимул к эволюции. Смартфоны играют важную роль в повседневной жизни человека. Актуальность выбранной темы исследования заключается в том, что в связи с возникающими изменениями, на рынке постепенно сформировался отдельный сегмент – мобильные приложения, специфика которого характеризуется тем, что разработка приложений должна проводиться с учетом особенностей мобильных устройств: отличиями интерфейса, различных размеров экранов, сенсорным управлением. Разработанное приложение будет пользоваться спросом пользователей, которые занимаются спортом, следят за здоровьем или просто интересуются подробностями пеших прогулок.

Целью исследования является разработка мобильного приложения StepsTracker (шагомер) в облачной среде визуальной разработки MIT App Inventor. Для достижения поставленной цели выделены следующие задачи:

- изучить теоретические основы среды разработки приложений MIT App Inventor;
- охарактеризовать функциональное назначение приложения StepsTracker;
- разработать и протестировать мобильное приложение;
- представить описание работы приложения и привести пример расчета показателей.

Изложение основного материала

App Inventor (App – сокращение от Application (приложение), Inventor – изобретатель) – облачная среда визуальной разработки приложений для операционной системы Android. Изначально создана в Google Labs, однако после закрытия лаборатории (15 октября 2011 года) была передана [Массачусетскому технологическому институту](#), впоследствии название было изменено на MIT App Inventor. Для работы в данной среде требуется владение элементарными основами алгоритмизации и наличие минимальных познаний языков программирования.

В MIT App Inventor разработка мобильных приложений осуществляется в два этапа. Первый – проектирование интерфейса пользователя в режиме «Дизайнер», второй этап – программирование компонентов приложения в режиме «Блоки». Для разработки мобильных приложений необходимо использовать ноутбук или настольный ПК, а тестирование и отладку производить на мобильном устройстве с установленным приложением MIT App Inventor Companion, позволяющим считывать QR- код созданного приложения на ПК для установки его на мобильное устройство.

С развитием технологической сферы смартфон позволяет пользователю не только оставаться на связи, но и проводить наблюдения за личной деятельностью. Отслеживать уровень физической активности поможет StepsTracker (шагомер) – приложение, позволяющее распознавать каждое изменение смартфона в пространстве для расчета количества пройденных шагов, дистанции (м), сожженных калорий и затраченного на это времени (с), предварительно указав показатели роста (см) и веса (кг) пользователя. Впоследствии статистика обрабатывается и сохраняется в базу данных, а пользователь получает необходимую информацию.

Мобильный шагомер будет исправно функционировать только при условии, что в смартфоне имеется акселерометр (датчик положения в пространстве). Встроенная микросхема позволяет отслеживать перемещения смартфона в пространстве, определять скорость и направление движения. Как известно, человек движется неравномерно: в процессе совершения каждого шага происходит как ускорение (в момент «отталкивания» ногой от земли), так и замедление (в ходе «приземления» на ступню). Именно амплитуду данных незаметных изменений отслеживают датчики, информацию с которых считывает программа, при подсчете шагов. К примеру, смартфон в кармане «движется» вверх-вниз вместе с пользователем, который куда-то направляется. Акселерометр фиксирует эти изменения (частоту и амплитуду), а потом приложение подсчитывает количество шагов и другие показатели. Когда пользователь находится в автомобиле график колебаний совсем не похож на колебания при ходьбе, соответственно, смартфон отличает ходьбу от поездки на машине. Иногда показания отличаются от реальных, но имеются минимальные погрешности.

При разработке мобильного приложения StepsTracker (шагомер) в MIT App Inventor были выбраны следующие компоненты (рис. 1). Размещение компонентов и их преобразование отображено на рис. 2.

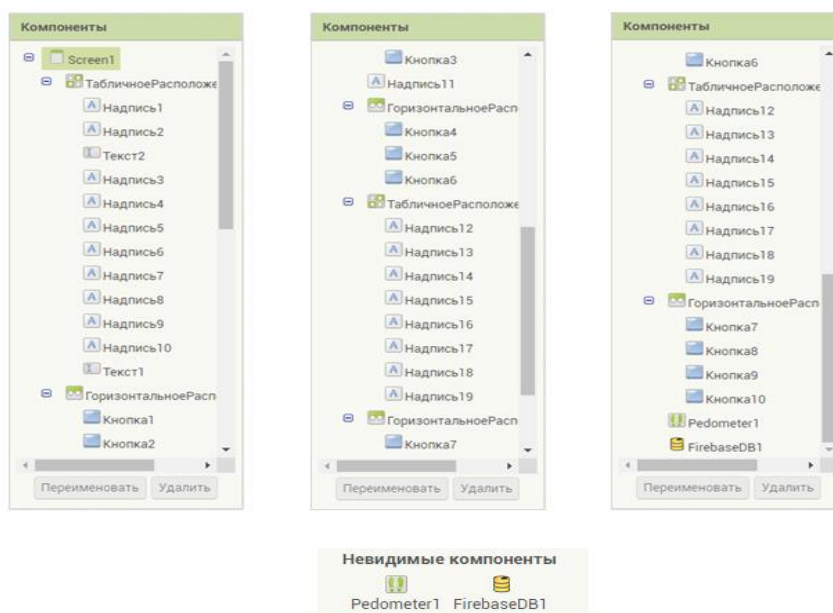


Рис. 1. Компоненты мобильного приложения StepsTracker



Рис. 2. Размещение и преобразование компонентов мобильного приложения StepsTracker

Внешний вид разработанного мобильного приложения при запуске на смартфоне и дальнейшее функционирование отображено на рис. 3.



Рис. 3. Разработанное мобильное приложение StepsTracker на смартфоне

Изначально работающей кнопкой является «Старт», а также поля для ввода параметров роста (см) и веса (кг), остальные кнопки – заблокированы. Для начала подсчета показателей после запуска приложения в специально отведенные поля необходимо ввести информацию о росте (см) и весе (кг) пользователя и нажать кнопку «Старт». Все кнопки, кроме «Стоп» – блокируются, после нажатия которой подсчет показателей останавливается, в отведенных полях выводятся полученные значения, и только кнопка «Стоп» становится заблокированной. При нажатии кнопки «Сбросить» все результаты будут удалены (без сохранения в базу данных). Нажатием кнопки «Добавить» значения сохраняются в базу данных, а при нажатии кнопки «Просмотр» – выводятся на экран в разделе (Активность), нажатие на кнопку «Удалить» – удаляет значения из базы данных. Кнопки «|<», «<», «>», «>|» отвечают за отображение полученных ранее и последующих показателей из базы данных. При разработке приложения была подключена база данных Firebase (рис. 4).

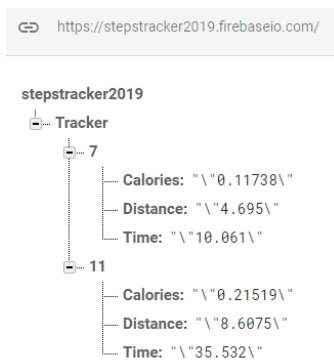


Рис. 4. База данных «Firebase» в приложении StepsTracker

В роли примера программирования компонентов в режиме «Блоки» приведен расчет по формулам основных показателей: пройденные шаги, дистанция, калории и время (рис. 5), для наглядности значения компонентов см. рис. 2.

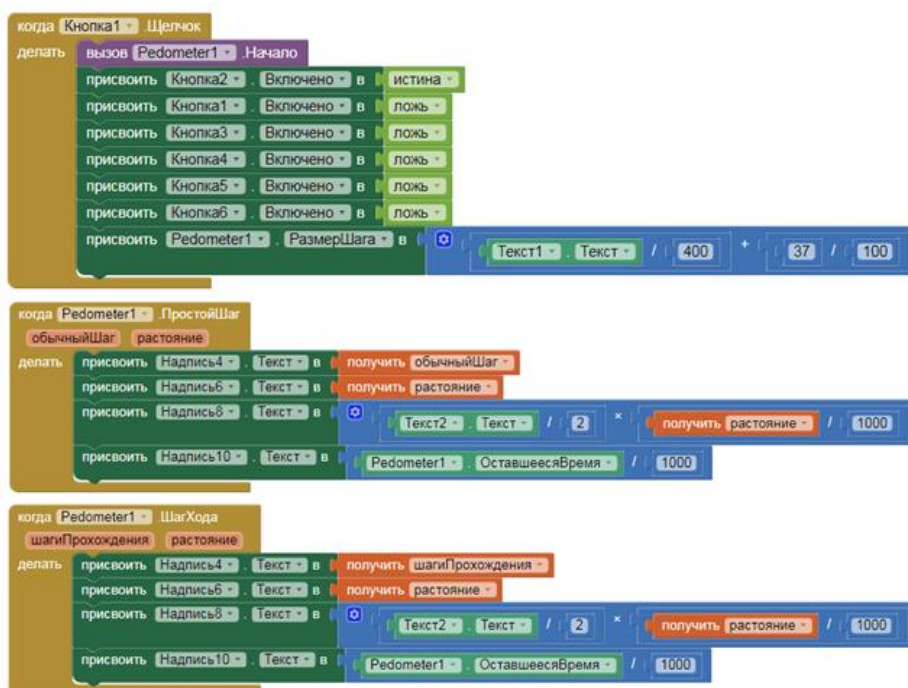


Рис. 5. Построение блоков приложения StepsTracker

Выводы

В результате исследования была изучена облачная среда визуальной разработки приложений для платформы OS Android MIT App Inventor. Разработано приложение StepsTracker (шагомер), которое на сегодняшний день является актуальным и полезным для пользователей, которые занимаются спортом, следят за здоровьем или просто интересуются подробностями пеших прогулок.

Литература

1. App Inventor [Электронный ресурс] / Интернет-ресурс. – Режим доступа : <http://appinvent.ru/f/uroki/AppInventor-Obzor.pdf>. – Загл. с экрана.

УДК 004.932

СИСТЕМА РАСПОЗНАВАНИЯ ЛЖИ ПО МИМИКЕ ЧЕЛОВЕКА НА ОСНОВЕ ВИДЕО

Лютый В.А., Секирин А.И.

Донецкий национальный технический университет
кафедра автоматизированных систем управления
E-mail: vlad11092000@mail.ru

Аннотация: Лютый В.А., Секирин А.И. Система распознавания лжи по мимике человека на основе видео. В статье приведено описание современных методов обработки видеoinформации для определения и распознавания факта лжи человека по его мимике. Произведен сравнительный анализ существующих методов, для решения поставленной задачи, а также выявлены основные проблемы, возникающие при обработке видеоданных для подобного рода задач.

Annotation: Luty V.A., Sekirin A.I. Deception detection system based on human facial expression by videoframes. The article describes the modern methods of processing video information to identify and recognize the fact that a person is lying by his facial expressions. A comparative analysis of the existing methods was performed to solve the set task, and the main problems that arise when processing video data for such tasks are identified.

Введение

В повседневной жизни ложью называют намеренную передачу информации с целью вызвать в другом человеке убеждение, которое сам передающий считает не соответствующим истине. Ложными могут быть факты и сведения, ошибки в которых могут нанести ущерб разного рода областей деятельности людей. Истинность или ложность информации может быть неточной из-за недостатка данных. Ложь приобретает особое значение когда речь касается политики, судебной процессии, медицины, рабочего процесса и т.д.

Для проведения анализа мимики человека на оценку достоверности информации эксперты пользуются правилами и мимическими картами, полученных с помощью психофизиологических исследований в области психологии человека, что является недостатком применения его в разных сферах деятельности, т.к. специализирующих экспертов в данной области не так много.

Первым техническим конкурентом стал всем известный полиграф. Использование полиграфа имеет свою достаточно длительную историю. Родоначальником полиграфа являются США, где проблеме его применения уделялось значительное внимание на различных уровнях (федеральном, региональном) и различными инстанциями, вплоть до конгресса США, президента, Верховного суда США и др. Статистика результатов работы полиграфа поражает: фактическая оценка работы (95%) превышает заявленную теоретическую (80%). То есть лишь 5-7 человек из ста способны пройти полиграф так, что результат будет «непонятным» [1].

Однако при всех достоинствах использования полиграфа, у него есть ряд недостатков и самые значительные из них – дорогостоящее оборудование (порядка 200 тыс. рублей), невозможность использования без осведомления субъекта допроса, необходим сертифицированный эксперт для анализа показаний устройства. В конце можно сказать, что

на равне с развитием компьютерной техники, роста вычислительных мощностей и методов обучения искусственного интеллекта и полиграф нельзя назвать технологией XXI века.

Постановка задачи

Проблеме обнаружения и распознавания лжи и обмана по внешним признакам с помощью обработки видеоряда посвящено множество работ, однако в целом она ещё далека от разрешения. Основные трудности заключаются в том, чтобы можно было распознать объект исследования (человек) независимо от его внешних особенностей (пол, национальность, возраст, физические и врожденные повреждения и т.д.), ракурс и условия освещенности при съёмке, а также при различных изменениях, связанных с причёской, одеждой, татуировками и т. д [2].

Распознавание признаков обмана по выражению лица является сложным психофизиологическим процессом и является крайне сложным для понимания необученного человека в этой области. Однако даже сертифицированные эксперты в области анализа физиогномики не всегда может правильно распознать признаки обмана и это может занимать достаточно большое количество времени, а это означает, что для систем автоматического распознавания данная задача является куда сложнее.

Цель исследования – провести анализ существующих математических подходов для распознавания и классификации лжи по мимике человека на видеокдрах.

Обобщенный алгоритм системы

Основной алгоритм автоматического распознавания лиц и мимики на изображениях видеопоследовательности состоит из следующих этапов:

1. Регистрация изображения.
2. Первичная обработка изображения с целью повышения эффективности работы алгоритмов распознавания.
2. Выделение лица на изображении.
3. Выделение частей лица.
4. Выделение ключевых точек на лице.
5. Отслеживание изменения контрольных точек с объектом на последовательности кадрах.
6. Классифицировать с помощью признаков полученные данные трекинга на предмет лжи.

Обработка изображения

Первичная или предварительная обработка изображения включает в себя удаление шумов, геометрические и цветовые преобразования. Источниками шума могут быть конструктивные недостатки средств регистрации изображений, плохое освещение сцены, механические воздействия на оборудование, положение объектов интереса, помехи в каналах передачи информации и др. Для удаления шумов можно применить усредняющие фильтры или фильтры, основанные на порядковых статистиках. К усредняющим фильтрам относятся, например, фильтры, основанные на вычислении среднего арифметического, среднего геометрического, среднего гармонического, среднего контргармонического. К фильтрам, основанным на порядковых статистиках, относятся, например, медианный фильтр, фильтр максимума, фильтр минимума.

Обнаружение и распознавание объектов в кадре

В первую очередь после получения очередного кадра, требуется выделить на нем объекты, представляющие интерес, а именно фигуры людей, и затем уточнить тип найденного объекта.

Для решения данной задачи существует множество различных подходов, а именно сегментация, детектирование признаков, категориальное распознавание и т.д., для дальнейшего анализа были выделены следующие группы методов: методы с применением сегментации, методы с сопоставлением с шаблоном и методы с извлечением признаков объектов.

Сегментация по цветовой составляющей

Методы поиска лиц одни по их близости к цветовому спектру человеческой кожи самые распространенные методы, так как они просты в реализации и отличаются низкими затратами на вычисления. Разные методы, из данной категории, используют разные цветовые системы, поэтому итоговый результат может меняться, относительно используемой системы. Основные цветовые системы в которых часто используют поиск лиц это RGB и TSL.

Но также встречаются методы с применением хроматических компонент цвета, которые показывают вполне не плохие результаты по обнаружению кожного покрова на изображении. При статистических сборах информации видно, в хроматических компонентах цвета (YCbCr) кожи разных людей, разных национальностей находится в двух областях, которые можно использовать как классификатор[4].

Главным недостатком методов этой группы является вычисление порога яркости и есть всегда есть большая вероятность, что он не идеальный.

Поиска объекта по шаблону

Данная группа методов работает по следующему принципу: требуется два компонента – исходное изображение и изображение-шаблон. Чтобы определить сходный участок, необходимо наложить шаблон на снимок и попиксельно двигать его, слева направо, сверху вниз. В каждом положении вычисляется метрика, которая показывает, совпадают картинки или нет. Для каждого положения шаблона метрика заносится в результирующую матрицу R. Каждое положение (x,y) в R содержит метрику совпадений.

После окончания сравнения лучшие совпадения находятся в глобальных минимумах (при использовании метода наименьших квадратов) или максимумах (при использовании метода корреляции или метода кросс-корреляции) [4].

Поиск объектов по шаблону хорошо применима при анализе сцен, в которых камера статична, а все экземпляры искомым объектов выглядят одинаково. На практике данные методы накладывают серьезные ограничения на ракурс съемки и на изменчивость объектов. Можно находить объекты конкретного типа, но не классы.

Категориальное распознавание объектов

Чтобы избежать вышеописанных проблем, современные системы распознавания объектов пользуются альтернативным подходом, а именно поиском фигур людей с помощью обученного классификатора. Использование классификатора подразумевает выделение некоторых характерных признаков на изображении, машинное обучение классификатора на наборе позитивных и негативных образцов, на которых заведомо присутствуют или отсутствуют объекты искомого класса, и применение классификатора к очередному изображению с целью поиска объектов на нем [2].

Среди наиболее популярных методов данной группы можно выделить следующие: метод гибкого сравнения на графах, метод главных компонент, сверточные нейросети [3].

Сравнительная характеристика алгоритмов, принадлежащих каждому из описанных выше подходов представлена в таблице 1.

Таблица 1 – Сравнение алгоритмов по перечню свойств

Свойство \ Алгоритм	Метод главных компонент	Сверточная нейронная сеть	Метод гибкого сравнения на графах
Устойчивость к положению лица	+	+	+
Устойчивость к изменениям освещения		+	+

Продолжение таблицы 1 – Сравнение алгоритмов по перечню свойств

Свойство \ Алгоритм	Метод главных компонент	Сверточная нейронная сеть	Метод гибкого сравнения на графах
Может использоваться для идентификации	+		+
Добавление нового лица без переобучения	+		+
Быстрое распознавание	+	+	

Анализ динамического изменения мимики лица в кадре

Следующим шагом является нахождение ключевых точек выделенных элементов лица. Рассмотрим существующие методы трекинга объектов.

1) Отслеживание точек может быть сформулировано как зависимость обнаруженных объектов, представленная точками во всех кадрах. Соответствие объектов в методах отслеживания точек обычно определяется с помощью комбинации ограничений:

- близость местоположения объекта от одного кадра к другому;
- максимальная скорость, ограничивающая окрестность вокруг объекта;
- плавность движения, значащая что направление и скорость объекта не меняются кардинально;
- общее движение, предполагающее что скорость объектов в малой окрестности должна быть одинаковой;
- жесткость, обозначающая что расстояние между любыми двумя точками на объекте останутся без изменений.

2) Представление объектов на основе простых геометрических форм вычисляемое движение неявно определяет область объекта, его ориентацию в следующем кадре, расположение объекта определяется с помощью оценочной модели движения объекта. На основе представления внешнего вида объекта эти методы можно разделить на две подкатегории, использующие шаблон вида объекта и использующие несколько видов объекта.

3) Отслеживание силуэта используется, когда требуется трекинг объектов сложной формы. Методы отслеживания контура можно разделить на две категории, а именно соответствия формы и эволюция контура. Методы соответствия формы похожи на методы шаблона вида объекта. Поиск выполняется путем вычисления схожести объекта с генерированной моделью из предполагаемых силуэтов, основанных на предыдущем кадре.

Методы эволюции контура последовательно развивают начальный контур в новую позицию в текущем кадре с помощью моделей пространства состояний или прямой минимизации функции энергии.

Сравнительная характеристика методов отслеживания представлена в таблице 2.

Таблица 2 – сравнительная характеристика методов трекинга объекта

Метод	Достоинства	Недостатки
-------	-------------	------------

Отслеживание точек	Подходят как для отслеживания очень маленьких объектов, так и для больших, для описания которых необходимо множество точек.	В контексте отслеживания объектов с использованием нескольких точек важной проблемой является кластеризация точек, необходимая для отличия между несколькими объектами и между объектами и фоном.
--------------------	---	---

Продолжение таблицы 2 – сравнительная характеристика методов трекинга объекта

Метод	Достоинства	Недостатки
Примитивные геометрические фигуры	Простота использования, низкая вычислительная стоимость, подходят для применения в реальном времени.	Части объектов могут быть оставлены за пределами определенной формы, в то время как часть фона может находиться внутри него.
Отслеживание силуэта	Гибкость при работе с большим разнообразием форм объектов.	Не позволяют отслеживать изменения топологии, а именно раскол региона или слияния.

Анализ методов отслеживания объектов показал, что в настоящее время остро стоит проблема обеспечения непрерывного трекинга. Важным фактором, влияющим на эффективность применения метода, является выбранный способ описания объекта.

Определить мимический признак можно на основе анализа нескольких ключевых точек. Например, на рис. 1 показаны комбинации точек бровей и рта (рис. 1, а – изображения бровей и рта; рис. 1, б – комбинация ключевых точек бровей и рта, соответствующая их изображениям; рис. 1, в – комбинация ключевых точек, соответствующая другим положениям рассматриваемых элементов на изображениях)[6].

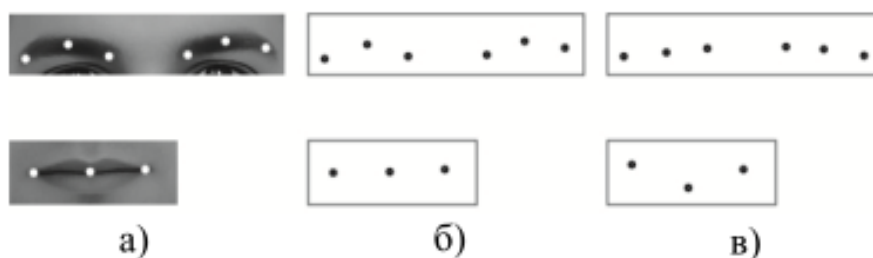


Рис. 1 - Ключевые точки элементов лица

Выделение ключевых точек осуществляется следующим образом:

1. Преобразование цветного изображения к полутоновому виду;
2. Преобразование из полутоновой формы к бинарному виду;
3. Применение к бинарному изображению градиентной маски;
4. Локализация ключевых точек. Переход от цветного изображения к полутоновому выполняется на этапе выделения лица.

Для получения бинарного изображения используется адаптивный порог. Применение градиентной маски к бинарному изображению позволяет получить контурное представление анализируемого элемента. Локализация ключевых точек заключается в определении заданного количества точек, лежащих на контуре элемента. Процесс выделения ключевых точек проиллюстрирован на рис. 2 (а - полутоновое изображение, б - бинарное изображение, в - применение градиентной маски, г - локализация ключевых точек) [6].

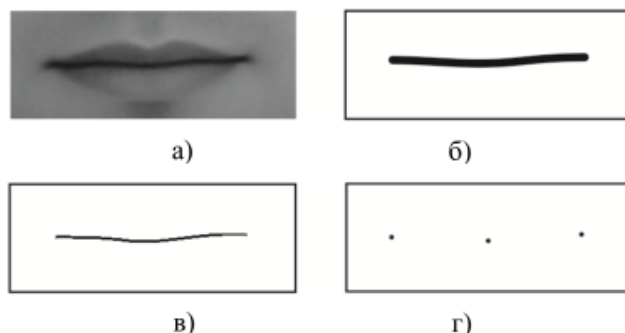


Рис. 3 - Выделение ключевых точек:

После выделения ключевых точек можно выполнить классификацию признаков лжи по известным правилам физиогномики [7]: Lip biting, Micro-expressions, Frequent swallowing, Sips of the tongue, Joint/merged expressions, Asymmetry in the face, Duration of the expression, Dilated pupils, Fewer facial movements, Blink rate.

Заключение

В статье исследуется процесс анализа мимики человека на предмет выявления в ней признаков лжи и обмана. Проведен сравнительный анализ существующих категорий методов решения поставленной задачи, выявлены достоинства и недостатки. Определены направления в решении подзадачи распознавания и классификации. В процессе исследования предметной области были выявлены существенные проблемы, которые могут возникнуть при обработке потока видеоданных, а именно сложность распознавания человека из-за индивидуальных особенностей внешности, изменения положения тела и лица на видеопотоке.

Литература

1. Culjak I., Abram D., Pribanic T., Dzapo H., Cifrek M. A brief introduction // 2012 Proceedings of the 35th International Convention MIPRO. Opatija, Croatia. 2012. P. 1725-1730.
2. Kalman R.E. A new approach to linear filtering and prediction problems // Journal of Basic Engineering, Vol. 82(1), 1960. P. 35-45.
3. Cristianini N., Shawe-Taylor J. An Introduction to Support Vector Machines and Other Kernel based Learning Methods. New York: Cambridge University Press, 2000.
4. Dalal N., Triggs B. Histograms of oriented gradients for human detection // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2005. Vol. 1. P. 886-893.
5. Amos B., Ludwiczuk B., Satyanarayanan M. Openface: A general-purpose face recognition library with mobile applications, CMU School of Computer Science, Pittsburgh, Pennsylvania, United States, CMU-CS-16-118, 2016.
6. Viola P., Jones M. Rapid Object Detection using a Boosted Cascade of Simple Features // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Kaii, Hawaii, USA. 2001. Vol. 1. P. 511-518.
7. Bay H., Ess A., Tuytelaars T., Van Gool L. Speeded-Up Robust Features (SURF) // Computer Vision and Image Understanding. June 2008. Vol. 110(3). P. 346-359.

УДК 004.93.2.72`1

АВТОМАТИЗИРОВАННЫЙ ПОДХОД К ВЫБОРУ РАДИОЛОГИЧЕСКИХ КАРТ ДЛЯ ЛУЧЕВОЙ ТЕРАПИИ

Ежело С.Е., Васяева Т.А.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: pesork2@gmail.com, vasyaeva@gmail.com

Аннотация:

Ежело С.Е., Васяева Т.А. Автоматизированный подход к выбору радиологических карт для лучевой терапии. В статье приведено описание методов обработки изображения рентген снимка для подбора радиологической карты с минимальным разбросом волн. Рассмотрены такие этапы решения поставленной задачи как выделение объектов на снимке, уточнение типов найденных объектов, отслеживание объекта на следующих снимках и выявление необходимости выбора нескольких карт, если таковы будут. Произведен сравнительный анализ существующих групп методов, для решения поставленной задачи, а также выявлены основные проблемы.

Annotation:

Yezhelo S.E., Vasyaeva T.A. Automated approach to the selection of radiological maps for radiotherapy. The article describes the image processing methods for the radiological map with minimal wave spread. Consideration of such stages, the solution of tasks, such as the selection of objects in the image, clarification of detected objects, tracking objects in the following images and identifying the need to select multiple maps, if any. A comparative analysis of the existing group methods was carried out to solve the set tasks, and the main problems were identified.

Введение

Одним из главных аспектов лечения онкологических заболеваний является лучевая терапия. Для того, чтобы начать лечение человека, врач топограф выявляет пораженные участки тканей человека на рентгене, так называемые «артефакты». В дальнейшем врач определяет тип онкологической патологии, а так же количество и тип излучения, которое направленно на уничтожения пораженных онкологией тканей. Для минимизирования ущерба по здоровым клеткам организма, были созданы так называемые «радиологические карты» - это карты, которые отображают по какой закономерности будут рассеиваться волны определённого типа (альфа, бэта, гамма). На данный момент подбор этих карт происходит вручную, что очень замедляет темп лечения пациента. Из-за чего лечение затягивается, и на здоровье пациента это отображается неудовлетворительно.

Принцип работы радиологических карт в лучевой терапии

Лучевая терапия — это направленное использование радиации для лечения новообразований и ряда неопухолевых заболеваний. При облучении происходит гибель больных клеток, что останавливает развитие болезни. Если не проводить лечение, больные клетки могут непрерывно расти, разрушая при этом здоровые клетки и распространяясь по всему организму.[6]

При планировании лечения, задействуется такой фактор как точечные излучения непосредственно в очаг пораженных клеток. Для минимизации ущерба здоровым тканям принято использовать стандартизированные радиологические карты для распределения

дозиметрических излучений. Благодаря этому, появляется возможность минимизировать урон, наносимый аппаратом с источником здоровым тканям пациента с онкологическим заболеванием. [6]

Облучение используется для лечения многих типов опухоли, и для многих пациентов оно является единственным методом лечения, который им необходим. Лучевая терапия может приносить уменьшение болей, симптомов сдавления здоровых органов, прекращать кровотечение.

Источник излучения можно подводить к больному участку несколькими способами, если источник:

- находится на расстоянии от тела пациента - облучение называется дистанционным;
- помещают в какую-либо полость - внутрисполостное;
- вводят непосредственно в больной участок в виде жидкости, проволоки, игл, зондов, а также в виде жидкости через рот, в сосуды и полости.

На данный момент ведущей системой, которая занимается 2-D планированием является

BRANYVISION. Varian Medical Systems - предназначены для лучевой терапии новообразований и заболеваний различной локализации при наличии показаний для данного вида лечения. Достоинства систем следующие:

- регистрация изображений;
- возможность регистрации изображений полученных методами КТ, МРТ;
- графическая верификация методом движущегося или расщепленного окна;
- получение изображений посредством DICOM 3.0 или напрямую с сервера ARIA;

Можно отметить следующие недостатки систем от Varian Medical Systems:

- стоимость этой системы остается невероятно высокой;
- доступна в основном только в виде дополнительной опции при покупке специализированного оборудования от производителя.

Постановка задачи

Наша задача разработать подход к выбору из заданных шаблонов радиологических карт, карту с минимальным разбросом волн.

Принцип работы заключается в том, что на снимке пациента (рис. 1) будет отображаться пораженный участок тела, после этого методом подбора, будет выбран набор шаблонов радиологических карт (рис. 2) с минимальным поражением здоровых тканей человека.

Каждая волна отмечена процентом получения облучения в процентах (70%, 80%, 90%, 100%), опухоль непосредственно должна располагаться в отметке 100%. Это так называемый «очаг», наша основная задача расположить его так, чтобы сопутствующий ущерб был минимизирован по отношению к здоровым тканям.

Этапы работы условно можно разделить на две части:

- 1) Определить пораженную область на снимке.
- 2) Путём использования методов обработки изображений, определить какой из шаблонов радиологических карт подходит к данному снимку.



Рисунок 1 – Пример снимка с пораженной областью

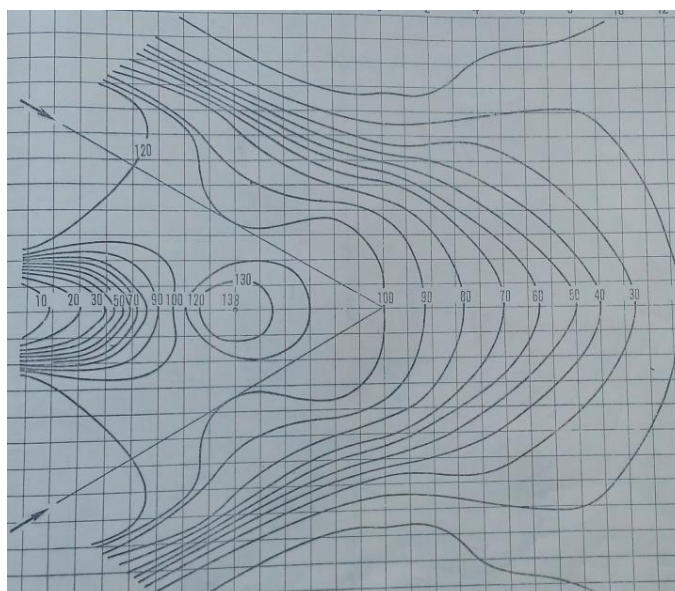


Рисунок 2 – Пример радиологической карты

Цель исследования – подбор радиологической карты с минимальным разбросом волн.

Обнаружение и распознавание объектов

В первую очередь после получения очередного изображения, требуется выделить на нем объекты, представляющие интерес, а именно очаг, и затем уточнить тип найденного объекта. Для решения данной задачи существует множество различных подходов, но наиболее оптимальны для неё это: метод главных компонент, детектирование признаков, метод наименьших квадратов.[2]

Детектирование объектов с использованием метода поиска объекта по шаблону

Данная группа методов работает по следующему принципу: требуется два компонента – исходное изображение и изображение-шаблон. Чтобы определить сходный участок, необходимо наложить шаблон на снимок и попиксельно двигать его, слева направо, сверху вниз. В каждом положении вычисляется метрика, которая показывает, совпадают картинки

или нет. Для каждого положения шаблона метрика заносится в результирующую матрицу R . Каждое положение (x, y) в R содержит метрику совпадений.[4]

Метод главных компонент (МГК)

Является одним из способов уменьшения размерности данных при минимальной потере информации. Данный метод сводится к вычислению собственных векторов и собственных значений ковариационной матрицы исходных данных [5]. Для применения метода главных компонент к люминесцентным изображениям таблица исходных данных формировалась следующим образом: изображение разбивалось на пиксели, каждый из которых нумеровался. [1] В свою очередь, каждый пиксель несёт в себе информацию об интенсивности трёх цветовых компонент: красной, зеленой и синей.

Метод наименьших квадратов

МНК заключается в нахождении таких (1) коэффициентов линейной зависимости, при которых значение функции двух переменных будет наименьшим.

$$F(a, b) = \sum n_i = 1(y_i - (ax_i + b))^2(1)$$

Иначе говоря, при определенных значениях a и b сумма квадратов отклонений представленных данных от получившейся прямой будет иметь минимальное значение. [3] В этом и состоит смысл метода наименьших квадратов.

Основные достоинства и недостатки работы каждого метода представлены в таблице 1.

Таблица 1 – Достоинства и недостатки методов группы распознавания изображений

Название методов	Достоинства	Недостатки
Метод главных компонент	<ul style="list-style-type: none"> - простота реализации; - низкое потребление памяти для хранения объектов; - простота добавления новых обученных фигур. 	<ul style="list-style-type: none"> - очень чувствителен к освещенности; - чувствителен к углу поворота объекта;
Детектирование объектов	<ul style="list-style-type: none"> - работает в условиях перекрытий; - ищет все имеющиеся на изображении экземпляры объекта за один проход. 	<ul style="list-style-type: none"> - сложность алгоритма растет экспоненциально с увеличением числа параметров модели.
Метод наименьших квадратов	<ul style="list-style-type: none"> - удобен в реализации; - доступность математических выводов 	<ul style="list-style-type: none"> - сложность вычислительной процедуры - чувствительность оценок к резким выбросам

Для того, чтобы выяснить целесообразность применения описанных методов в системе анализа, было произведено сравнение с точки зрения точности распознавания, времени обработки одного изображения, вычислительной сложности и объема обучающей выборки.

Полученные результаты сравнения представлены в таблице 2.

Таблица 2 – Сравнение методов обнаружения объекта

Метод	Объем	Время обработки	Вычислительная	Точность
-------	-------	-----------------	----------------	----------

	обучающей выборки, шт.	одного изображения, мс	сложность	распознавания
Главных компонент	29303	100	+	~80%
Детектирование объектов	26405	100	+++	~85%
Метод наименьших квадратов	28140	100	+	~90%

Примечание: количеством знаков «+» отмечается степень различия для критерия.

Таким образом, метод наименьших квадратов существенно проще при проведении вычислительной процедуры и дает хорошие по статистическим свойствам оценки. Этот метод наиболее подходит для выполнения поставленной задачи.

В процессе изучения существующих методов были проведены экспериментальные исследования для выбора наиболее эффективного метода с точки зрения точности распознавания.

Таблица 3 – Сравнение методов распознавания

Название Метода	Достоинства	Недостатки
Метод водораздела	Удовлетворительно локализует наличие текстуры отличной от фона	Чрезмерная сегментация
Метод нормальных разрезов	Хорошо локализирует наличие текстуры отличной от фона. Хорошая модель представления изображения	Не обеспечивают локализацию объектов. Не выделяет изменение яркости и поворот текстуры Большое время выполнения
Метод оценки изменения формы спектра	Удовлетворительно обеспечивает локализацию объектов, локализирует дефекты	Не локализует дефекты, образованные поворотом Больше время выполнения

Заключение

В статье рассмотрена проблема подбора радиологических карт для лучевой терапии. Проведен сравнительный анализ существующих категорий методов решения поставленной задачи, выявлены достоинства и недостатки. Определены направления в решении подзадачи распознавания. В процессе исследования предметной области были выявлены существенные проблемы, которые могут возникнуть при обработке изображения, а именно сложность в точности подбора карты для очага для минимизации поражения не пораженных клеток.

Литература

1. Donoser, M. and Bischof, H. Efficient Maximally Stable Extremal Region (MSER) Tracking. //
2. Anop MF, Katueva EV, Mihalichuk V.I. Particle swarm algorithms in the image processing problem // Science and Education. - 2015. - № 1.
3. Vodolazsky I. A., Egorov A. S., Krasnov A. V. Genetic algorithm and its most common methods of implementation

4. Bay H., Ess A., Tuytelaars T., Van Gool L. Speeded-Up Robust Features (SURF) // Computer Vision and Image Understanding. June 2008. Vol. 110(3). P. 346-359.

5. Е.Е. Борисов, О.Д Функо, On the task of finding an object in the image // Science and Education – 2014. - №2

6. N.A. Lichman, N.G. Kukva, N.G. Semikoz Radiation therapy of primary and metastatic tumors 2018

УДК 004.67

ИСПОЛЬЗОВАНИЕ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ДЛЯ ФОРМАЛИЗАЦИИ ЗАДАЧ ПРИНЯТИЯ РЕШЕНИЙ

Барабанщиков О. Е., Савкова Е. О.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: barvius@yandex.ru

Аннотация:

Барабанщиков О.Е., Савкова Е.О. Использование имитационного моделирования для формализации задач принятия решений. В статье предложено использование имитационного моделирования для формализации многокритериальной задачи принятия решения о развитии предприятия на реальном примере. При этом процесс функционирования предприятия представлен в виде системы массового обслуживания. В качестве каналов обслуживания рассматриваются варианты размещения разного количества оборудования, а с помощью моделирующего алгоритма получены оценки этих вариантов по двум критериям.

Abstract:

Barabanshikov O. E., Savkova E.O. Using simulation modeling to formalize decision making tasks. The article proposes the use of simulation modeling for the formalization a multi-criteria task of making a decision on the development of an enterprise using a real example. At the same time, the enterprise operation process is represented as a queuing system. As service channels, the placement different amounts of equipment is considered, and using the modeling algorithm, estimates of these options are obtained by two criteria.

Актуальность задачи

При модернизации и расширении производства, обновлении и замене оборудования, внедрении новых технологий руководителю предприятия приходится принимать решение о целесообразности таких действий. Под целесообразностью действий понимается их оценка по различным параметрам, таким как минимальные затраты, максимальная прибыль, повышение качества обслуживания или выпускаемой продукции, улучшение условий труда, снижение трудоемкости работ и т.п.

Подобные задачи имеют следующую характерную особенность:

- модель, описывающая множество допустимых решений, объективна, но качество решения оценивается по многим критериям;
- для выбора наилучшего варианта решения необходим компромисс между оценками по различным критериям;
- в условиях задачи отсутствует информация, позволяющая найти такой компромисс. Следовательно, он не может быть определен на основе объективных расчетов.

Имитационное моделирование основано на воспроизведении с помощью компьютера развернутого во времени процесса функционирования системы с учетом взаимодействия с

внешней средой. Области применения методов имитации чрезвычайно широки и разнообразны. В разрезе данной статьи будет рассматриваться использование имитационного моделирования к проблеме формализации задач принятия решений [1], а именно получения оценочных показателей различных вариантов действий (альтернатив). При этом функционирование предприятия представляется в виде системы массового обслуживания (СМО).

СМО – класс математических схем, разработанных в теории массового обслуживания и различных приложениях для формализации процессов функционирования систем, которые по своей сути являются процессами обслуживания [2]. Основными элементами СМО являются входной поток заявок, входной поток обслуживаний, очереди заявок, ожидающих обслуживания, каналы обслуживания и выходной поток обслуженных заявок и заявок, которым по тем или иным причинам в обслуживании отказано. Характерным для работы таких систем является случайное появление заявок на обслуживание и завершение обслуживания в случайные моменты времени, то есть стохастический характер их функционирования. Таким образом, в любом элементарном акте обслуживания можно выделить две основные составляющие: ожидание обслуживания заявкой и само обслуживание заявки.

В результате возникает проблема составления алгоритмов на машине с последовательной обработкой таких процессов, которая состоит в том, что при моделировании необходимо отслеживать множество моментов, которые в реальном времени происходят параллельно. При реализации программы на компьютере процессор выполняет определённую последовательность операций, имитируя функционирование системы во времени. Одновременно с этим накапливаются и обрабатываются численные характеристики процесса, которые в дальнейшем будут использованы для оценки качества работы системы.

Решение проблемы

Рассмотрим решение реальной задачи размещения оборудования в условиях морского порта.

В морском порту имеется 3 причала, на каждом из которых могут работать несколько грузовых кранов. Эти грузовые краны выполняют как разгрузочные, так и погрузочные работы.

В порт через промежутки времени τ , имеющие равномерный закон распределения с $m=2$ часа и $\sigma=0,5$ часа, прибывают корабли 3-х типов с вероятностями 0.2; 0.3; 0.5.

Корабли 1 типа могут разгружаться и загружаться на всех 3-х причалах.

Корабли 2 типа - на 1 или 2 причалах

Корабли 3-го типа только разгружаются и только на 3 причале.

Время разгрузки или погрузки кораблей одним краном подчиняется нормальному закону с параметрами: $m_1=10$ ч, $\sigma_1=3$ ч; $m_2=15$ ч, $\sigma_2=5$ ч; $m_3=20$ ч, $\sigma_3=6$ ч.

Определить, сколько необходимо поставить кранов на каждом причале, чтобы корабли не ждали разгрузки более 3 часов. Статистику следует собрать за 10 дней работы порта. Корабли, которые к данному моменту не были обслужены, можно не учитывать.

Критериями для определения оптимального количества кранов на каждом причале являются минимальная стоимость и минимальное время ожидания обслуживания.

В результате получена многокритериальная задача принятия решения, где альтернативой является количество кранов, устанавливаемых на каждом причале. Матрица полезностей такой задачи может быть получена с помощью моделирующего алгоритма системы разгрузки и погрузки кораблей в порту.

Основной моделирующий алгоритм использует метод особых состояний [3] и изображен на рисунке 1.

В начале алгоритма выполняется инициализация массива времени освобождений каналов rg , текущего времени t , количества обслуженных кораблей i и общего времени

ожидания обслуживания (блок 1). В блоке 2 организуется цикл 10 дневной обработки заявок. В теле цикла увеличивается текущее время на время поступления заявки (блок 3), определяется тип прибывшего судна (блок 4), после чего в блоке 5 анализируется очередь кораблей. Если очередь не пустая, то проверяется возможность обслуживания судна из очереди, для этого в цикле 6 перебираются заявки в очереди, определяется свободный канал (блок 7). Если тип корабля соответствует типу канала, то судно обслуживается (блок 9) и удаляется из очереди (блок 10). Аналогичные действия выполняются, для только что пришедшей заявки, т.е. проверяется возможность ее обслуживания (блоки 11 и 12), заявка обслуживается (блок 13) по исходу «да» блока 12, или добавляется в очередь (блок 14), и счетчик кораблей увеличивается на 1 (блок 15). В процессе моделирования для большей наглядности можно организовать вывод промежуточных значений.

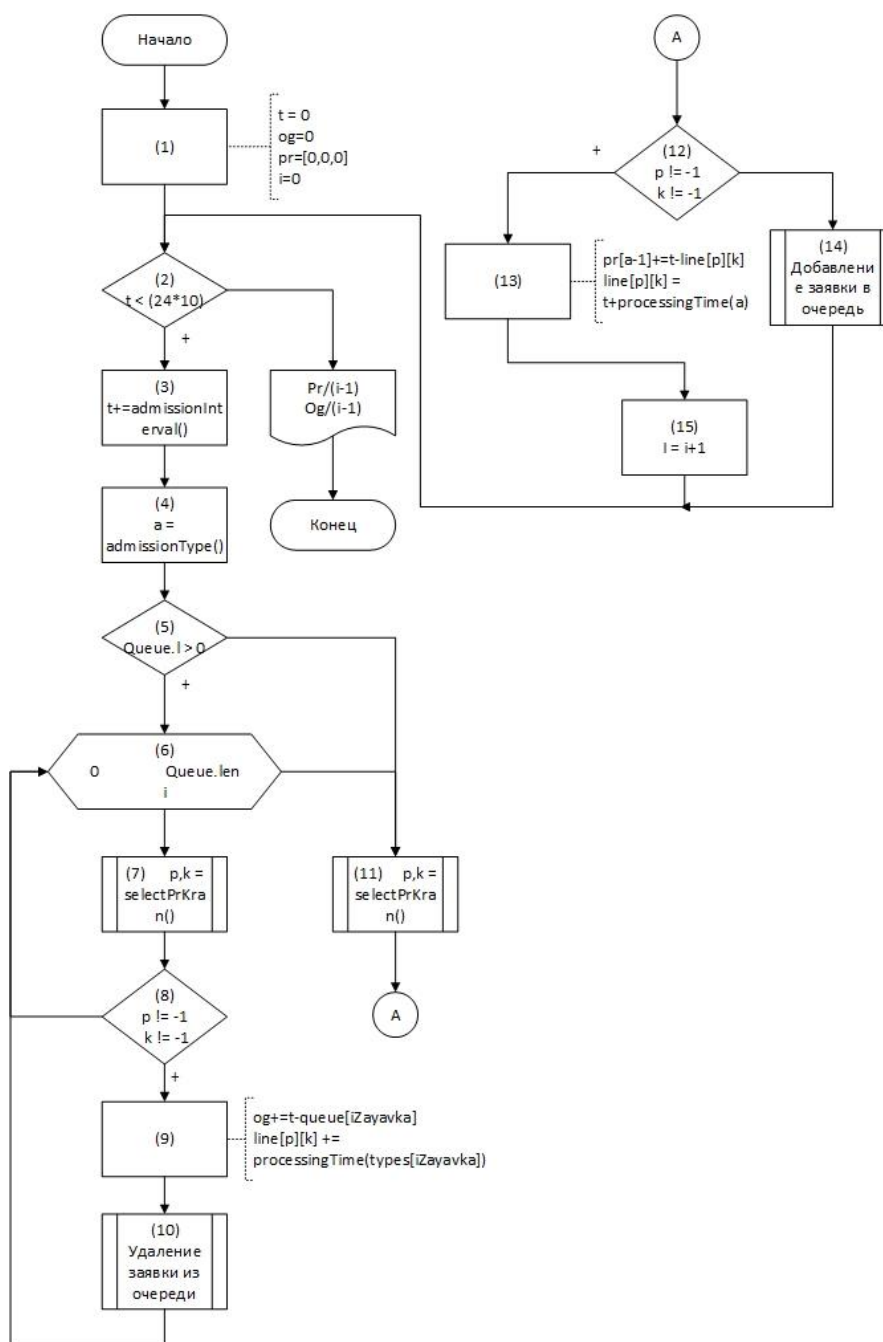


Рисунок 1 – Основной моделирующий алгоритм

Анализ результатов

Результаты моделирования представлены диаграммами использования каналов и постановки заявок в очередь. Каждая вертикальная линия – некоторый момент времени, а двойная вертикальная – единица времени, после которой корабли больше не поступают, а обслуживаются те, что остались в очереди. Числовые обозначения на шкале очереди – количество кораблей, ожидающих обслуживания. Также следует заметить, что на графике отображено окончание моделирования, поэтому имеются занятые каналы в начале временной шкалы. График отображающий процесс загрузки каналов изображен на рисунке 2.

Для уменьшения размера диаграммы было использовано всего 5 кранов на трех причалах.

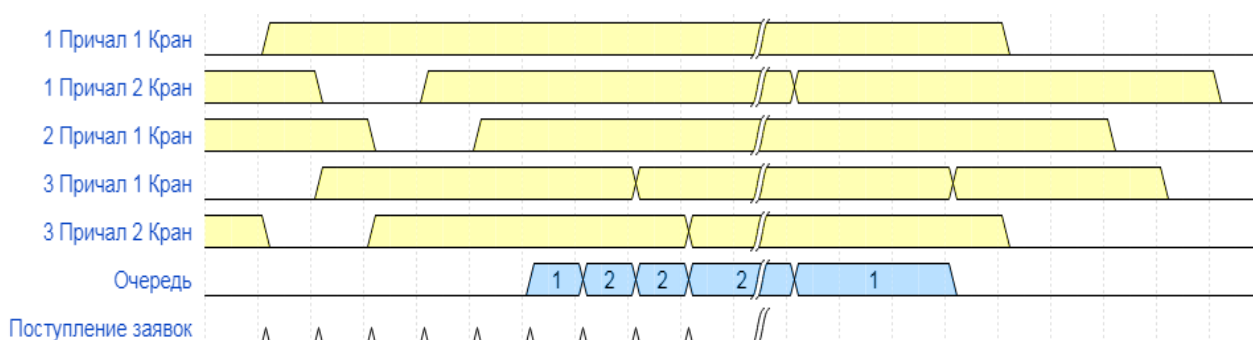


Рисунок 2 – Диаграмма загрузки каналов

Разработанный алгоритм был использован для моделирования системы с различным количеством кранов на причалах с целью определения их оптимального количества, при этом критериями отбора вычисляемые системой параметры: среднее время ожидания обслуживания, т.е. простой заявок. Если время простоя превышает 3 часа, то такой вариант расстановки кранов на причалах (альтернатива) не рассматривается и исключается из перечня альтернатив. Стоимость устанавливаемых кранов рассчитывается по формуле:

$$\sum_{i=1}^3 C_i K_i$$

где C – стоимость крана на i -том причале;

K – количество кранов на причале.

В формуле учитывается, что на каждом причале устанавливаются краны разного типа.

При расчетах использовались следующие данные:

$C1 = 2,2$ млн.руб; $C2 = 2,7$ млн.руб; $C3 = 3,1$ млн.руб;

Результаты моделирования были сведены в таблицу:

Количество кранов на причалах	Стоимость кранов, млн. руб	Среднее время простоя кораблей на причалах, час		
		первом	втором	третьем
A1	28,4	1,05	1,08	1,16
A2	28,8	0,97	0,51	2,02
A3	31,5	1,42	1,31	2,75
A4	27,9	1,08	1,22	1,29
A5	31	1,56	2,27	3,63
A6	32	1,36	1,12	3,16

A1 – 2 крана на первом причале, 2 крана на втором причале и 6 кранов на третьем причале;

A2 – 2 крана на первом причале, 1 крана на втором причале и 7 кранов на третьем причале;

A3 – 2 крана на первом причале, 2 крана на втором причале и 7 кранов на третьем причале;

A4 – 3 крана на первом причале, 1 крана на втором причале и 6 кранов на третьем причале;

A5 – 3 крана на первом причале, 1 крана на втором причале и 7 кранов на третьем причале;

A6 – 1 крана на первом причале, 3 крана на втором причале и 7 кранов на третьем причале;

Альтернативы A3, A5 и A6 являются доминируемыми и могут быть отброшены

Для окончательного выбора оптимального варианта размещения кранов на причалах применялись различные методы решения многокритериальных задач: метод главного критерия, методы свертки, метод целевого программирования, метод уступок, метод гарантированного результата [4]. Результат четырех методов одинаковый – оптимальным вариантом расстановки кранов на причалах для эффективного обеспечения работы порта является альтернатива A2. Окончательный результат должен приниматься ответственным лицом.

Выводы

В рамках данной статьи рассмотрен пример оценки альтернативных решений многокритериальной задачи с помощью представления процесса функционирования предприятия в виде системы массового обслуживания и разработки алгоритма, моделирующего работу такой системы. Показана целесообразность такого подхода, давшего возможность рассчитать матрицу полезности с дальнейшим использованием различных методов решения многокритериальных задач для выбора эффективного варианта действий.

Литература

1. Милая А.С., Савкова Е.О. Формализация задачи принятия решения об оптимизации работы по обслуживанию клиентов в условиях неопределённости. Электронные информационные системы, №1 (12) 2017 г. – М., АО «НТЦ ЭЛИНС». – с. 82-93.
2. Советов Б.Я., Яковлев С.А. Моделирование систем. Учебник для бакалавров. Юрайт-издат, 2013 г. – 343 с.
3. Бусленко Н.П. Моделирование сложных систем. М.: Наука, 1978. – 399 с.
4. Методы решения многокритериальных задач [Электронный ресурс]. – Режим доступа: <http://studfiles.net>

УДК 004.021

УПРАВЛЕНИЕ ТРАНСПОРТНЫМИ ПОТОКАМИ С ИСПОЛЬЗОВАНИЕМ АДАПТИВНЫХ АЛГОРИТМОВ

Довидович В.С.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: vladik.dvdivch@gmail.com

Аннотация:

Довидович В.С. Управление транспортными потоками с использованием адаптивных алгоритмов. Выполнен анализ алгоритмов управления транспортными потоками применимых к автоматизированным системам управления дорожным движением. Приведены принципы, обеспечивающие обеспечения максимальной эффективности управления и надежности функционирования системы. Предложен алгоритм адаптивного регулирования транспортного потока на четырехстороннем пересечении.

Annotation:

Dovidovich V.S. Contemporary trends in machine learning technology. In this article We analyzed traffic flow control algorithms applicable to automated traffic management systems. Also Provides guidelines to maximum management efficiency and reliability of the system. In addition, algorithm for adaptive regulation of traffic flow at a four-way intersection.

Введение

В настоящее время во многих крупных городах весьма остро стоит проблема транспортных заторов. При этом исследования [1] показывают, что потенциал существующих улично-дорожных сетей используется далеко не полностью. Повышение пропускной способности улично-дорожных сетей может быть достигнуто за счет внедрения автоматизированных систем управления дорожным движением (АСУДД). При внедрении АСУДД достигается улучшение следующих показателей [2, 3]: время в пути транспортных средств снижается на 10–15 %; количество общих транспортных остановок сокращается на 20–40 %; расход топлива снижается на 5–15 %, количество вредных выбросов в атмосферу сокращается на 5–15 %; повышается безопасность дорожного движения.

Актуальность задачи

Совершенствование и развитие автоматизированных систем управления дорожным движением имеет особое значение для любого крупного мегаполиса. Данные о существующей интенсивности движения могут быть получены в результате обследований транспортных и пешеходных потоков, а о перспективных значениях этого показателя – методом компьютерного моделирования с использованием транспортных моделей.

Необходимость применения адаптивных методов определяется уровнем загрузки элементов улично-дорожной сети. При принятии решений о необходимости использования методов адаптивного управления в качестве порогового значения уровня загрузки, как правило, используется уровень 80 %. При этом для центральных районов города рассматривается транспортная ситуация, характерная для дневного пика интенсивности буднего дня, для периферийных районов учитывается ситуация, обусловленная массовым выездом за город – вечер пятницы летнего периода.

Требования к системе

Общегородская (АСУДД) должна быть построена на принципах, обеспечивающих максимальную эффективность управления и надежность функционирования системы. В общесистемной области такими принципами являются децентрализация управления и гибкость структуры связей в системе. Децентрализация управления – это отказ от постоянной передачи управляющих воздействий из центра к периферийным контроллерам, имеющей место во всех отечественных АСУДД. При децентрализованном управлении связь центра с контроллером осуществляется только в случаях: сопровождение спецтранспорта в режиме «зеленой улицы», при отработке контрольных режимов, в аварийных ситуациях.

В области технологического обеспечения основными принципами, реализуемыми в общегородской АСУДД, являются гибкость и адаптивность.

Возможность гибкого изменения пространственной структуры АСУДД повышает эффективность работы системы, что явствует из следующего примера. Во время дневного пика перекрестки должны работать в режиме координированного управления с общим циклом регулирования. В период спада интенсивности (вечер, ночь, выходные дни) целесообразно сохранить режим координации на магистрали, а соседние перекрестки либо объединить в новые небольшие районы координации, либо просто перевести в режим локального управления. Отсутствие возможности гибкого конфигурирования районов координации может привести к неоправданному росту задержек транспорта.

Алгоритмы адаптивного управления

Алгоритмы адаптивного управления локального применения могут быть основаны с помощью комбинирования различных методов адаптивного управления длительностью циклов и фаз, последовательностями фаз, методов промежуточными тактами, а также методами обеспечения приоритетного пропуски. Однако задача разработки универсального алгоритма лишена смысла ввиду того, что такая задача является в крайней степени многофакторной, поэтому ее решение нецелесообразно с точки зрения трудозатрат. Верной стратегией в этом случае является разработка алгоритмов локального управления, учитывающих особенности конкретных пересечений.

Для регулирования движения на пересечениях, как правило, применяются методы регулирования, основанные на адаптивном управлении длительностью циклов и фаз. Из этой группы наиболее просты в реализации метод поиска разрыва и метод разъезда очереди.

Гипотеза исследования звучит следующим образом: применение комбинации метода поиска разрыва и метода разъезда очереди способно обеспечить более высокий уровень обслуживания на регулируемых пересечениях с приведенной интенсивностью движения 300–500 легк. авт./час, приходящейся на каждую полосу движения, по сравнению с жестким регулированием.

В настоящее время дорожные контроллеры работают по принципам, основанным на таких понятиях как «кольцо» и «барьер» [5].

Кольцо состоит из двух или более последовательно рассчитанных и индивидуально подобранных фаз конфликтных направлений, организованных так, чтобы они появлялись в заданном порядке.

Барьер или линия совместимости – это базисная точка, в которой увязываются в заданной последовательности оба кольца контроллера. Два барьера не дают возможности последовательного выбора и установки конфликтных фаз, расположенных в разных кольцах. Оба кольца пересекают барьер одновременно для выбора и установки фаз на другой стороне барьера.

Для моделирования работы алгоритма управления транспортными потоками на пересечении было решено использовать двух кольцевую диаграмму с одним барьером. При этом было выбрано четырехстороннее пересечение, как наиболее распространенное в городских условиях. Структура двойного кольца показана на рисунке 1.

Двух кольцевой контролер содержит два взаимоувязанных кольца, связанных ограничительным барьером, причем одновременно в каждом из колец может задаваться своя фазировка. Таким образом, структура кольца представляет собой порядок (последовательность) обслуживания фаз движения и указывает совместимость фаз друг с другом.

Максимальное количество фаз, используемых в двух кольцевом контроллере с одним барьером, равно 8. Каждая из фаз управляет сигналом светофора (желтый, зеленый, красный). Восемь фаз необходимо для управления 8-ю направлениями движения на перекрестке (четыре прямых направления и четыре поворота налево).

Двух кольцевой контролер содержит два взаимоувязанных кольца, связанных ограничительным барьером, причем одновременно в каждом из колец может задаваться своя фазировка. Таким образом, структура кольца представляет собой порядок (последовательность) обслуживания фаз движения и указывает совместимость фаз друг с другом.

Максимальное количество фаз, используемых в двух кольцевом контроллере с одним барьером, равно 8. Каждая из фаз управляет сигналом светофора (желтый, зеленый, красный). Восемь фаз необходимо для управления 8-ю направлениями движения на перекрестке (четыре прямых направления и четыре поворота налево).

Фазы с номерами 1-4 включены в первое кольцо, фазы с номерами 5-8 включены во второе кольцо. Оба кольца действуют независимо друг от друга, за исключением того, что пересечение барьера происходит в обоих кольцах одновременно.

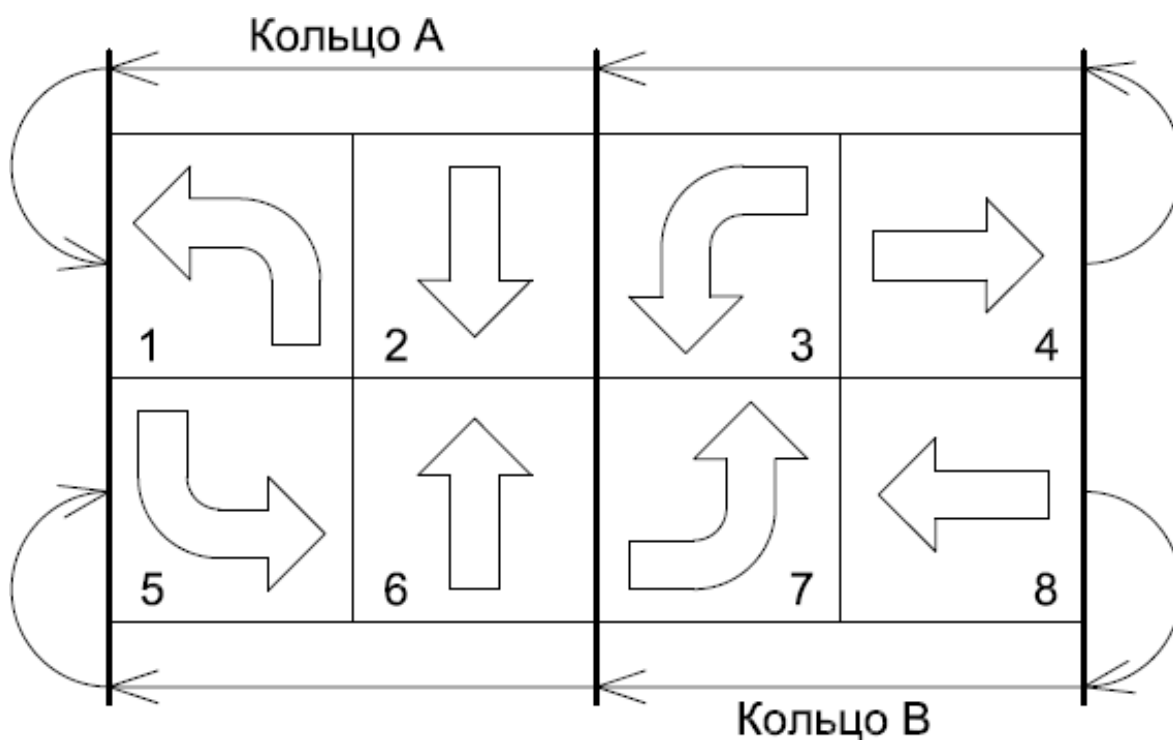


Рисунок 1 - Структура двойного кольца

Если все фазы назначены правильно, контроллер в ходе своих операций не будет предоставлять право проезда по двум конфликтным направлениям одновременно.

Разрабатываемый алгоритм

Разрабатываемый алгоритм адаптивного регулирования транспортных потоков на четырехстороннем пересечении основан на двух методах: методе поиска разрыва и методе разъезда очереди.

Метод поиска разрыва хорошо зарекомендовал себя при работе с потоками сравнительно малой интенсивности, в то время как метод разъезда очереди хорошо справляется с транспортными потоками высокой интенсивности, а также при неравномерном прибытии транспортных средств к регулируемому пересечению. Алгоритм, основанный на комбинации этих методов, способен в равной степени эффективно справляться с потоками транспортных средств с различной интенсивности и неравномерностью прибытия.

Разрабатываемый алгоритм управляет длительностью разрешающих сигналов в каждой фазе цикла регулирования. Управляющими параметрами являются:

- t_k^{\min} – минимальное значение длительности фазы k ;
- t_k^{\max} – максимальное значение длительности фазы k ;
- $t_{\text{ЭК}}$ – экипажное время.

Минимальная длительность фазы рассчитывается с учетом необходимости пропуска транспортных средств в количестве, определяемом расстоянием от стоп-линии до контролируемого сечения. Максимальная длительность основного такта должна обеспечивать допустимое время ожидания разрешающего сигнала на направлениях, движение которых запрещено в фазе k . Экипажное время определяется продолжительностью периода проезда транспорта от контролируемого сечения до стоп-линии.

Алгоритм работает следующим образом:

1. Перед началом основного такта производится регистрация количества транспортных средств, ожидающих разрешающего сигнала в соответствующем направлении $N_{\text{ож}}$.
2. В начале основного такта производится расчет минимального времени разрешающего сигнала:

$$t_k^{\min} = f(N_{\text{ож}}). \quad (1)$$

После этого производится сравнение полученного значения со значением длительности разрешающего сигнала, минимально необходимой для пересечения пешеходами проезжей части $t_k^{\text{пеш}}$. Из двух значений выбирается наибольшее.

3. С началом основного такта фиксируется прохождение автомобилями контролируемого сечения, и каждый автомобиль, проходящий через сечение в период отработки основного такта, продлевает его минимальную длительность на величину экипажного времени $t_{\text{ЭК}}$, в том случае, если до окончания основного такта остался резерв времени меньший $t_{\text{ЭК}}$. Тем самым транспортное средство обеспечивает свой проход через стоп-линию во время текущего такта.

$$t_k^{\text{тек}} = t_k^{\min} + t_{\text{ЭК}}, \quad (2)$$

где $t_k^{\text{тек}}$ – текущая длительность разрешающего сигнала основной фазы, с.

4. Длительность экипажного времени рассчитывается для каждого транспортного средства, пересекшего контролируемое сечение.

$$t_{\text{ЭК}} = f(V_{\text{авт}}), \quad (3)$$

где $V_{\text{авт}}$ – мгновенная скорость транспортного средства в контролируемом сечении, с.

Для регистрации скорости транспортных средств предполагается установка либо датчика измерения скорости транспортного средства, либо последовательная установка двух датчиков присутствия.

5. Основной такт заканчивается, если достигнута его максимальная длительность t_k^{\max} или в контролируемом сечении в течение экипажного времени не появился ни один автомобиль после истечения минимальной длительности, то есть в транспортном потоке появился разрыв.

Расчет t_k^{\max} самая сложная задача в работе данного алгоритма, так как описание зависимости этого параметра от других параметров потока вызывает затруднения ввиду неочевидности связи. В первом приближении предлагается использовать максимальную длительность соответствующего основного такта для жесткого алгоритма управления.

Выводы

В итоге можно сделать вывод, что на ранних этапах внедрения общегородской АСУДД следует первостепенное внимание уделять локальному адаптивному регулированию. Дальнейшее развитие общегородской АСУДД неизбежно приведет к формированию стратегического компонента, который объединит автономные локальные пересечения в единую сеть, способную обеспечить еще более высокий уровень обслуживания по сравнению с совокупным эффектом от организации повсеместного локального адаптивного управления.

Литература

1. Бродский Г. С., Айвазов А. Р. Автоматизированное управление дорожным движением в городской среде // Мир дорог. 2007. № 26. С. 2–3
2. Бродский Г. С., Рыкунов В. В. Поехали! АСУДД — мировой опыт и экономический смысл // Мир дорог. 2008. № 32. С. 36–39.
3. ГНПО АГАТ. <http://www.agat.by> (дата обращения: 16.06.2010).
4. Врубель Ю. А. Исследования в дорожном движении: учеб. -метод. пособие к лабораторным работам. Минск: БНТУ, 2007. 178 с.
5. Кременец Ю. А. Технические средства организации дорожного движения: учебник для вузов. М.: Академкнига, 2005. 279 с.
6. Левашев А. Г., Михайлов А. Ю., Головных И. М. Проектирование регулируемых пересечений: учеб. пособие. Иркутск: Изд-во ИрГТУ, 2007. 208 с.
7. LHOVRA: a traffic signal control strategy for isolated junction. Swedish national road administration, publication. 1991. 15E. 186 p.
8. SCATS Adaptive Traffic Control System, Brochure No. TC-21XX-8/08-250, TransCore. 2008. 8 p.

УДК 004.04

ОБНАРУЖЕНИЕ УГРОЗ БЕЗОПАСНОСТИ В СИСТЕМЕ КОНТРОЛЯ И УПРАВЛЕНИЯ ДОСТУПОМ ПРИ ПОМОЩИ ДЕРЕВЬЕВ РЕШЕНИЙ

Юрьев Н. Н., Васяева Т. А.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: kolyayurev@mail.ru, vasyaeva@gmail.com

Аннотация:

Юрьев Н. Н., Васяева Т. А. Обнаружение угроз безопасности в системе контроля и управления доступом при помощи деревьев решений. Рассмотрена проблема безопасности в системах контроля доступа. Предложен и реализован метод обнаружения угроз безопасности. Проведены необходимые эксперименты.

Annotation:

Yurev N. N. Vasyaeva T. A. Detection of security threats in the access control system with the help of decision trees. The problem of security in access control systems is considered. A method for detecting security threats is proposed and implemented. The necessary experiments were carried out.

Актуальность

В целях наиболее полного обеспечения безопасности охраняемого объекта (ОО), помимо аппаратной части системы контроля и управления доступом (СКУД) (которая выполняет функции физической защиты территории ОО от несанкционированного доступа) и программного модуля работающего напрямую с контроллерами и турникетом СКУД, (дающего необходимый функционал для удаленного управления и настройки вышеуказанных устройств), необходима система реагирования на внештатные и малозаметные события в системе, угрожающие безопасности ОО.

Применительно к СКУД, под угрозой безопасности понимается нанесение вреда предприятию и его работникам вследствие взлома, проникновения или чрезвычайных ситуаций, которые возникают вследствие физических и поведенческих уязвимостей.

В мировой практике, в качестве решения задачи данного рода применяют системы класса Security Information and Event Management(SIEM).

Системы SIEM самостоятельно не способны предотвращать инциденты, как и не имеет встроенных защитных функций. Предназначение таких систем состоит в анализе и корреляции данных, поступающих от различных иных систем, таких как Data Leak Prevention (DLP), Intrusion Detection System (IDS), межсетевых экранов, антивирусов, активного сетевого оборудования, СКУД, сканеров уязвимостей, и т.д., а также уведомлять об инцидентах потенциально являющихся угрозой безопасности[3].

Для эксплуатации таких систем необходимо наличие квалифицированного специалиста (эксперта), который обеспечит контроль непрерывности сбора событий, управление правилами корреляции, а также будет корректировать, и обновлять правила в соответствии с изменениями.

Цель работы – реализовать предложенный ранее[2] математический аппарат по обнаружению угроз и провести эксперименты.

Исследование

В качестве объекта исследования применяется разработанная ранее структура СКУД [1], которая представлена на рис. 1.

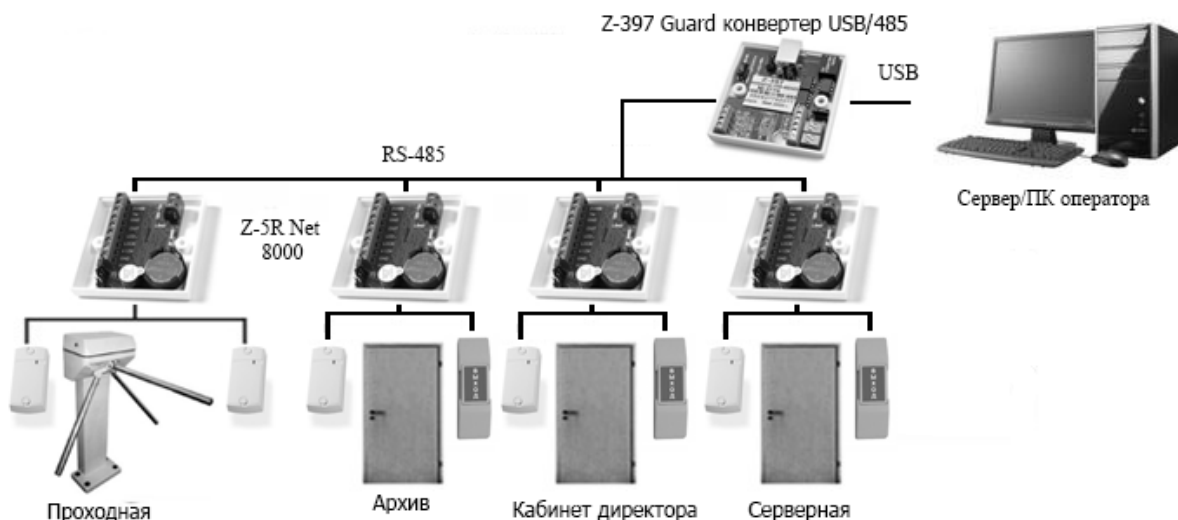


Рисунок 1 – Структурная схема СКУД

Данные о событиях, произошедших в СКУД, посредством специализированного программного обеспечения (ПО), изымаются из памяти контроллеров и помещаются в базу данных (БД). Таким образом, система формирует так называемые - списки событий (рис. 2). Однако само по себе наличие в БД подобного списка не дает полной картины произошедшего в ОО и рабочему персоналу весьма затруднительно визуально определить потенциальную угрозу безопасности по мелким, часто разрозненным во времени событиям.

Дата	Время	Точка доступа	Направление	Событие	Объект	
					Таб. №	
2008-05-14	00:09:09	1	выход	Зарегистрирован проход, санкционированный с кнопки.	не определен	
2008-05-14	06:09:36	5	вход	Зарегистрирован проход.	5525	Веретехин А. А.
2008-05-14	06:19:19	4	вход	Доступ запрещен. Неизвестный код пропуска.	не определен	
2008-05-14	11:09:07	1	(нет)	Связь с точкой доступа восстановлена.	не определен	

Рисунок 2 – Пример данных списка событий

В рамках данного исследования, для автоматизации обнаружения угроз предлагается реализация механизма обнаружения угроз посредством реализации метода случайных лесов[4], являющегося методом машинного обучения (МО). Математическая модель представляет собой ансамбль бинарных деревьев решений (алгоритм CART), построенных на обучающей бутстреп выборке [2]. Процесс МО требует достаточного количества данных, соответствующих как нормальному режиму работы системы, так и аномальным ситуациям. Обучение алгоритма осуществляется на наборах данных предварительно прошедших этапы фильтрации, корреляции событий и принимающих на выходе вид наборов параметров. Каждый такой набор эксперт должен соотнести с соответствующему ему уровню предполагаемой угрозы. В конечном итоге данные должны принять следующий вид рис. 3.

WorkT	Firstive	LevelA	LevelA	Object	TryAcc	Count	TryObj	TryLev	Eqfail	FailObj	FailLev	Multipl	Power	Power	Power	OpenC	OpenC	OpenC	FailDoi	FailDoi	FailDoi	Insidef	Y
1	x	2	1	6	x	1	6	1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Y_0
1	x	2	1	8	x	1	8	1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Y_0
x	x	x	x	0	x	3	13	2	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Y_1
x	x	x	x	0	x	3	14	2	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Y_1
0	x	x	x	0	x	4	2	2	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Y_1
1	x	x	x	0	x	0	x	2	1	2	1	x	x	x	x	x	x	x	x	x	x	x	Y_1
0	x	x	x	0	x	0	x	2	1	2	1	x	x	x	x	x	x	x	x	x	x	x	Y_2
0	x	x	x	0	x	0	x	2	1	13	2	x	x	x	x	x	x	x	x	x	x	x	Y_2

Рисунок 3 – Пример обучающих данных

Обучающий набор подается на вход системы обнаружения угроз и согласно заданным параметрам алгоритма (таких как размерность ансамбля и бутстреп выборки, а также процента прироста информации при разбиении в узле, при котором он принимает вид листа, тем самым соотнося себя с определенным классом угрозы) строится ансамбль из бинарных деревьев решений.

Зачем в уже обученный алгоритм подаются тестовые данные для определения класса потенциальной угрозы, используя метод голосования, который заключается в том, что в качестве окончательного решения принимается значение, чаще всего встречающееся в ансамбле деревьев.

Для проведения практических экспериментов была разработана программа на языке C#, позволяющую регулировать исходные параметры алгоритма и получать на выходе точность классификации.

Эксперимент

В рамках исследования были произведены ряд экспериментов с размерностями ансамблей и обучающей бутстреп выборки, для выявления их значений при которых точность классификации достигнет приемлемого значения, с ошибкой не более 1%.

Для обучения алгоритма были промоделированы события возникающие в СКУД, соответствующие как внештатным ситуация там и нормальному режиму работы. Каждому набору были заданы уровни потенциальной угрозы. Таким образом, изначальная обучающая выборка составила 950 наборов данных.

Результаты проведенного эксперимента представлены ниже на таблицах 1 и 2.

Таблица 1 – Точность классификации без отсечения ветвей, %.

Бутстреп выборка \ Ансамбль	100	200	400	600	800	950
10	81,77	90,06	93,51	95,72	97,51	97,38
100	90,88	94,61	97,65	98,76	99,72	99,72
500	91,71	95,17	98,62	99,31	99,72	99,72
750	92,54	94,89	98,48	99,31	99,72	99,72
1000	92,4	94,75	98,2	99,17	99,59	99,72
1500	92,27	94,34	98,48	99,31	99,59	99,72

В таблице 1 представлены данные полученные в ходе тестирования алгоритма без отсечения ветвей, т.е. узел становится листом только в том случае если все наборы данных в нем принадлежат строго к одному классу. В теории это могло бы привести к переобучению алгоритма и падению точности классификации, однако при тестировании этого не произошло. Делая вывод из результатов представленных в таблице выше, видим что максимальная точность классификации достигает 99.72%, что соответствует ошибке 0.28% находящейся в пределах нормы. Также можно сделать вывод, что для достижения максимальной доступной точности классификации, достаточно ансамбля из 100 деревьев и бутстреп выборке равной 800 наборов из исходных 950.

Еще одним интересным фактом является уменьшение точности классификации при количестве деревьев в ансамбле равным 1500 и малых бутстреп выборках. Из чего делаем вывод что дальнейшее увеличение ансамбля не приводит к улучшению результата и в итоге при бутстреп выборке равной обучающей, приводит к тем же результатам, что и при ансамбле из 100 деревьев.

В таблице 2 представлены результаты классификации с отсечением ветвей в деревья, при восьмидесяти процентном приросте информации в узле. В данном случае видно, что максимальная точность классификации значительно уменьшилась и не превышает 92%. Однако для достижения максимально возможной точности понадобился гораздо больший ансамбль состоящий из 1000 деревьев и бутстреп выборке равной исходной. Также наблюдается похожая тенденция роста ошибки классификации при ансамбле из 1500 деревьев.

Таблица 2 – Точность классификации с отсечением ветвей при 80% прироста информации в узле.

Бутстреп выборка Ансамбль	100	200	400	600	800	950
10	76,8-81,35	81,97-85,05	85,64-88,05	86,33-88,2	87,94-89,64	89,43-91,05
100	81,49-83,46	85,91-87,02	87,85-88,95	88,54-89,78	89,5-90,19	89,92-91,16
500	82,46-83,43	86,46-87,43	88,95	89,36-89,64	90,47-90,88	90,61-91,32
750	82,87-83,29	86,6-87,15	88,81-89,23	89,36-89,64	89,5-90,88	90,75-91,57
1000	82,87-83,56	86,74-87,29	89,09-89,23	89,36-89,5	90,75-90,88	91,3-92
1500	83,15-83,56	86,33-87,02	88,95-89,05	89,5	90,61-90,88	90,75-91,3

Выводы

Рассмотрена актуальная задача обнаружения угроз системы контроля и управления доступом. Реализован и на практике протестирован метод случайных лесов применительно к задаче обнаружения угроз, показавший удовлетворительные результаты классификации.

Литература

1. Юрьев Н.Н. Система контроля и управления доступом / Н.Н. Юрьев, Т.А. Васяева, С.Д. Бельков, Н.С. Суббота // Информатика, управляющие системы, математическое и компьютерное моделирование. – 2017. – № 7. – С. 601-604.
2. Юрьев Н.Н. Обнаружение угроз безопасности в системе контроля и управление доступом / Н.Н. Юрьев, Т.А. Васяева // Информатика, управляющие системы, математическое и компьютерное моделирование. – 2018. – № 8. – С. 126-129.
3. Кириллов В.А. Система сбора и корреляции событий (SIEM) как ядро системы информационной безопасности / В.А. Кириллов, А.Р. Касимова, А.Д. Алёхин // Казань: Вестник Казанского технологического ун-та. 2016. № 13. С. 132–134.
4. Чистяков С. П. Случайные леса: обзор //Труды Карельского научного центра Российской академии наук. – 2013. – №. 1.

УДК 004.932

РАЗРАБОТКА МОДЕЛИ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ ФОРМИРОВАНИЯ ЦЕН В ИНТЕРНЕТ-МАГАЗИНЕ

Лисничук В.Е., Землянская С. Ю.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

Email: lisnichuk.v.e@gmail.com

Аннотация

Лисничук В.Е., Землянская С. Ю. Разработка модели интеллектуальной системы формирования цен в интернет-магазине. Проанализированы методы прогнозирования спроса для формирования цен на товары в зависимости от сезонности и других влияющих факторов. Исследованы факторы, влияющие на цены на товары, сформулирован критерий оптимальности цены на товар, предложена модель интеллектуальной системы расчета оптимальной цены на товары с учетом истории продаж и цен конкурентов.

Annotation

Lisnichuk V.E., Zemlyansky S. Y. Development of model of an intellectual system of pricing in online store. Demand forecast methods for pricing for goods depending on seasonality and other influencing factors are analysed. The factors influencing goods prices are investigated, the criterion of formation of the best price of goods is formulated, the model of an intellectual system of calculation of the best price of goods taking into account a sales history and the prices of competitors is offered.

Постановка проблемы

Задача формирования цен на товары актуальна в торговой области деятельности.

При ценообразовании решаются такие задачи как учет спроса на товар в определенный период, учет факторов, влияющих на формирование цены, а также учет цен конкурентов.

Наиболее важной задачей является формирование оптимальной цены на товары, при которой будет получена максимальная прибыль от продаж.

Формирование цены представляет собой сложный многогранный процесс. Процесс ценообразования представлен на рисунке 1.1. Необходимую для установления цены информацию следует изучить, проанализировать. Недостаток информации, так же как и ее чрезмерность, затрудняют решение проблемы.



Рис. 1.1 – Процесс ценообразования

Основной целью любого магазина является получение максимальной прибыли, а этого можно достичь за счет увеличения количества продаж. Для достижения поставленной цели, прежде всего, необходимо изучить предпочтения, потребности покупателей и основные факторы, влияющие на формирование цены, на основании чего разработать модель информационной системы, формирующей оптимальные цены на товары интернет-магазина.

Для того чтобы разработать подобную систему, необходимо сформулировать критерии, по которым формируется цена на товар, и исследовать методы, которые могут быть использованы в подобной системе, разработать модель формирования цен.

Рассмотрим интернет-магазин, в котором представлена некоторая номенклатура товаров $T_1...T_n$. Каждому товару необходимо определить цену $Ц_1...Ц_n$ такую, чтобы прибыль от продаж была максимальной и было продано максимально возможное количество товаров

$$\sum_{i=1}^m (\Pi_i - P_i) \rightarrow \max \quad (1)$$

$m \rightarrow n$

где n – количество видов товаров,

m – количество проданных видов товаров,

Π_i – продажи i -го вида товара,

P_i – закупочная цена i -го вида товара.

Продажи можно рассчитать по формуле:

$$\Pi_i = K_i \cdot Ц_i \quad (2)$$

где

K_i – количество проданного товара i -го вида,

$Ц_i$ – цена товара i -го вида.

Обзор исследований по теме

Цена на товар на определённом рынке не является постоянной величиной даже в пределах небольшого срока. Цены подвержены изменениям и колебаниям в зависимости от целого ряда факторов, начиная от политических и макроэкономических событий и заканчивая модой и погодой. Некоторые из этих факторов могут быть спрогнозированы, другие же носят вероятностный характер или вовсе непредсказуемы. В связи с этим при ценообразовании важно понимать, какие факторы оказывают влияние на определение цены, в какой мере производитель может использовать позитивные и нивелировать влияние негативных факторов.

На начальной стадии формируем цену по методу «Издержки плюс». Данный метод предполагает расчет цены продажи посредством прибавления к цене производства, цене закупки, хранения материалов и сырья фиксированной дополнительной величины прибыли [4]. Дополнительную величину прибыли устанавливает продавец. Уровень добавления суммы к себестоимости товара, устраивающий продавца, может быть не принят покупателем.

Для того чтобы установить оптимальную цену на товары, нужно спрогнозировать цену в зависимости от сезонности и других влияющих факторов. На основании начальной стоимости и прогнозирования можно сделать вывод, верно ли установлена цена или она нуждается в корректировке.

Существуют такие методы прогнозирования:

- 1) Метод простой средней;

- 2) Метод скользящего среднего;
- 3) Метод экспоненциального сглаживания;
- 4) Метод "Хольта-Уинтерса";
- 5) Метод Авторегрессии;
- 6) Нейронные сети.

Далее рассмотрим каждый метод прогнозирования в отдельности.

1) Метод простой средней.

Является простейшим из подобных методов, используется вычисление по формуле "простого среднего". Прогноз цены на товар на следующий период при этом способе вычисляется как среднее арифметическое показателей цены за все предыдущие периоды.

Недостатком этого метода является его высокая "консервативность" – устаревшая информация о прежних продажах помешает проявиться последним тенденциям спроса по такой цене [5].

2) Метод скользящего среднего

Более оперативно реагирует на изменение цены. Расчет при этом производится не на основании данных за весь срок наблюдения, а за несколько последних периодов.

Интересной вариацией метода является расчет скользящей средней по определенным сезонам. Такой метод может подойти товарам, обладающих ярко выраженной сезонностью [4].

3) Метод экспоненциального сглаживания

В отличие от предыдущих методов расчетов "по среднему", дающих лишь очень приближенные результаты, позволяет добиться более точного прогноза при использовании моделей "экспоненциального сглаживания" и "экспоненциального сглаживания с трендом".

В первом методе последний прогноз оптимальной цены корректируется на основе ошибки прогноза, допущенной в последнем периоде. При втором методе расчетов (называемом еще методом "двойного экспоненциального сглаживания") учитываются данные с трендами, благодаря чему данный метод может использоваться даже для среднесрочного прогнозирования [4].

4) Метод "Хольта-Уинтерса"

Многие продукты имеют тенденцию роста или падения продаж, особенно когда они производятся впервые или когда появляются конкурирующие товары. Для некоторых продуктов существенны сезонные изменения уровня продаж, поэтому для прогноза цен на товар целесообразно учитывать конкретный характер тенденции и сезонных колебаний. На основе модели Хольта Уинтерс (Винтерс, Winters) создал свою прогностическую модель, которая учитывает экспоненциальный тренд и аддитивную сезонность [5].

Чтобы получить прогноз цены, в этом методе необходимо правильно подобрать три параметра. Для этого можно использовать как специальные алгоритмы, так и ограничиться простым перебором.

5) Метод Авторегрессии

Эта методика позволяет провести очень подробный анализ имеющихся данных, выявить любые тенденции и отсеять случайные влияния, в результате получить более совершенный прогноз. Однако, в отличие от предыдущих методов, подбор множества параметров – довольно трудоемкую нетривиальную задачу. Прогнозирование с использованием модели авторегрессии опирается на предыдущие значения продаж. Зависимость в случае авторегрессии предполагается линейная, то есть прогноз представляет собой сумму продаж за предыдущие дни с некоторыми коэффициентами, которые являются постоянными и определяют параметры модели авторегрессии. Сколько дней (периодов, в общем случае) таких продаж из прошлого мы будем брать, чтобы попытаться спрогнозировать будущие продажи по назначенной цене называется порядком модели авторегрессии [4].

б) Нейронные сети

Следует отметить, что чем более сложные методы прогнозирования используются, тем труднее их практическое применение и тем выше вероятность возникновения ошибок. Анализ огромных объемов информации, подбор оптимальных параметров, оперативный учет колебаний рынка – все это порой находится на пределе человеческих возможностей. Наиболее перспективным в решении этой проблемы является использование алгоритмов "нейронных сетей".

В этой методике программа, реализующая нейронную сеть определенной структуры, после предварительного обучения способна самостоятельно найти лучшее решение, кроме того, "нейронные сети" способны учесть скрытые тенденции и создать достоверный прогноз в такой нестабильной ситуации, где ранее прогнозирование считалось вообще невозможным [5].

Динамическое ценообразование можно организовать с помощью искусственных нейронных сетей. Для обучения такой сети данные по уровню спроса, в зависимости от дня недели и от времени суток, берутся за прошлый период. Основное преимущество нейронной сети – это возможность обучаться и получать данные самостоятельно. В отличие от традиционных моделей спроса, модели, построенные на нейронных сетях, не делают каких-либо предварительных предположений о взаимосвязи между различными факторами, эти отношения извлекаются из самих данных. Они способны вывести значение из сложных или неточных данных и могут быть использованы для моделирования отношений, которые слишком сложны. Эта способность нейронных сетей делает их хорошим кандидатом для моделирования спроса при динамическом ценообразовании [1].

Нейронные сети позволяют решать задачи, с которыми не могут справиться традиционные методы, они способны успешно решать задачи, опираясь на неполную, зашумленную, искаженную информацию.

Рассмотрим схему организации динамического ценообразования с использованием нейронной сети (рис1.2).



Рис. 1.2. - Схема организации динамического ценообразования с применением нейронной сети

Компания в процессе своей деятельности осуществляет закупку товаров у поставщиков по оптовым ценам и реализует их населению по розничным. При этом формируется валовой доход предприятия, определяемый выручкой от реализации товаров и услуг за вычетом затрат на оплату стоимости полученных от поставщиков товаров. Компания стремится максимизировать свою чистую прибыль, которая при прочих фиксированных условиях, в том числе налоговых ставках, зависит от величин торговых наценок на товары. Существуют и другие факторы, например, покупательский спрос, конкурентоспособность товаров и скорость товарооборота, которые также влияют на величину прибыли.

Нейронная сеть получает данные от различных источников (цены у конкурентов, уровень спроса, цену оптовой закупки товара) анализирует их и формирует оптимальную цену на товар.

По итогам проведенного обзора было принято решение использовать метод прогнозирования цен с использованием нейронных сетей.

Вывод

Были исследованы различные методы формирования и прогнозирования цены на товары. Принято решение создать систему, используя метод «Издержки плюс» и метод прогнозирования «Нейронные сети». С помощью этой системы будет формироваться оптимальная цена на товары с учетом цен конкурентов, предпочтений и потребностей покупателей, а также с учетом сезона, жизненного цикла товара.

Литература

5. Использование нейронных сетей для динамического ценообразования [Электронный ресурс] – Режим доступа: URL: <http://edrf.ru/article/19-04-16>
6. Нейронные сети: на пороге будущего [Электронный ресурс] – Режим доступа: URL: http://www.neuroproject.ru/articles_dak_nn.php
7. Ценообразование [Электронный ресурс] – Режим доступа: URL: <https://ru.wikipedia.org/wiki/>
8. Методы прогнозирования цен [Электронный ресурс] – Режим доступа: URL: <https://works.doklad.ru/view/LqjyMIaIGtA/all.html>
9. Прогнозирование [Электронный ресурс] – Режим доступа: URL: <https://fnow.ru/articles/prognozirovanie-sprosa>

УДК 004.93.14

АВТОМАТИЗИРОВАННАЯ СИСТЕМА СЕГМЕНТАЦИИ ЛЕГКИХ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННОЙ СЕТИ

Пашков А.С. Васяева Т.А.

Донецкий национальный технический университет

Кафедра автоматизированных систем управления

Email: Lokhidy@gmail.com, vasyaeva@gmail.com

Аннотация:

Пашков А.С., Васяева Т.А. Автоматизированная система сегментации легких с использованием нейронной сети. Приведено описание архитектуры свёрточной нейронной сети, а также выделены основные этапы предобработки снимков. Исследован процесс работы нейронной сети U-NET и выделены основные этапы работы системы. Рассмотрены принципы, применяемые для обучения нейронной сети.

Annotation:

Pashkov A.S., Vasyaeva T.A. Automated lung-segmentation system based on a neural network. A description of the architecture of the convolutional neural network is given, and the main stages of image preprocessing are highlighted. The work process of the neural network U-NET is investigated and the main stages of the system operation are highlighted. The principles used to train the neural network are considered.

Введение

Медицинские рентген-снимки является одним из основных способов диагностики заболеваний. Информация, находящаяся в этих снимках, позволяет обнаружить заболевания легких на ранних стадиях (туберкулёз, эмфизема, патология дыхательных путей), а также нести данные о результатах лечения пациента. Получаемые снимки делятся на два основных типа. К первым относятся снимки, которые делаются при прохождении профосмотра. Получаемые изображения в большинстве случаев оказывается низкого качества, в виду регулярного износа оборудования, а также огромного потока пациентов. На данном этапе, врач может определить первичные показатели патологий, без внешних симптомов – болезни отростков бронхов, установление признаков запыления легких и т.д. Снимки второго типа делаются уже при постановлении определенного диагноза и делаются на более качественном оборудовании. Здесь уже можно определить не только наличие заболевания (эмфизема), но и установить степень заражения.

В среднем, врач-рентгенолог и врач-радиолог качественно могут обработать около 200 и 50 рентгенограмм соответственно. В крупных городах, особенно в период учащения заболевания пневмонией или в городах с предприятиями с опасными условиями труда (шахты, фабрики, заводы), поток снимков может существенно превысить указанные выше цифры, что сказывается на качестве их обработки. Современные системы компьютерно-медицинской диагностики позволяют упростить задачу врачам. Одной из таких задач является задача сегментации – выделения области интереса.

Постановка задачи

Способов решения задачи сегментации легких большое количество, но конкретного метода на сегодняшний день не существует, поскольку различные подходы дают качественный результат, но в разной степени. Применение технологий нейронных сетей в медицине относительно молодой проект и не всё аспекты до конца исследованы. В ходе

данного исследования будет использована модификация свёрточной нейронной сети U-Net, созданная непосредственно для решения задач в медицине.

Цель – исследовать архитектуру свёрточной нейронной сети и выделить основные моменты процесса обучения.

Предварительная обработка

Для обучения нейросети для установления классификатора требуется подготовить два набора данных – обучающая и тестовая выборка. Постановка задачи классификации ставится следующим образом: существует некоторое множество объектов с известным классификатором и объекты с неизвестной принадлежностью к классу, для которых требуется распределение.

Перед обучением нейронной сети, изображения необходимо предварительно обработать. В ходе тестирования основным набором методов будет являться:

- преобразование в полутон;
- стандартизация;
- метод CLАНЕ (Contrast Limited Adaptive Histogram Equalization);
- гамма-преобразования.

Возможным вариантом стоит рассматривать метод НЕ (Histogram equalization). Отличия заключаются в том, что метод НЕ охватывает всё изображение, а метод CLАНЕ разделяет изображение на отдельные сегменты – плитки, в которых строится собственная гистограмма. Выбор метода зависит от результатов предварительного тестирования и ручной оценки качества.

Архитектура нейронной сети

В качестве архитектуры была выбрана архитектура U-NET. Данная архитектура была создана в 2015 году с целью анализа биомедицинских изображения. Структура U-NET представлена на рисунке 1.

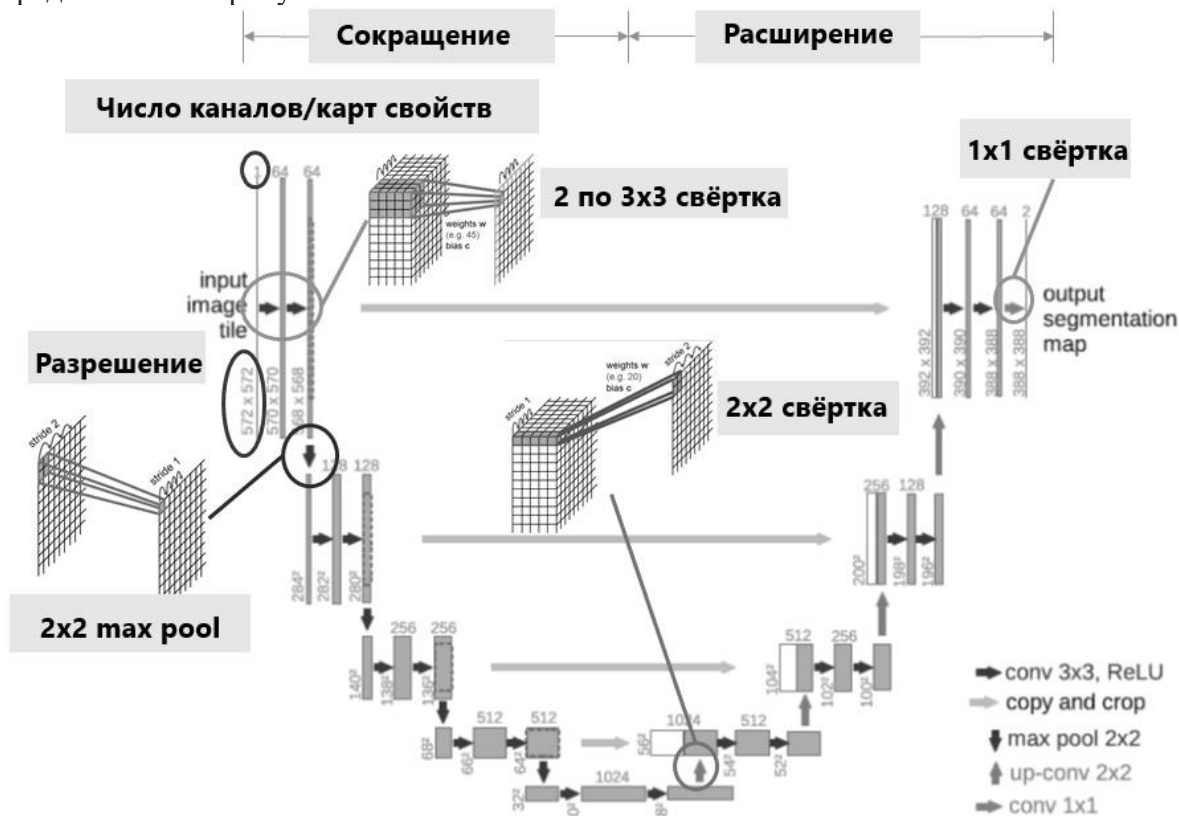


Рисунок 1 – Архитектура U-NET

Основной принцип заключается в повторном применении двух свёрток 3×3 с инициализацией ReLU (передаточная функция - выпрямитель) и операция максимальной конкатенации для уменьшения разрешения. На каждом этапе-итерации число карт-свойств удваивается [1].

На последнем слое используется единичная свёртка для сопоставления вектора свойств с желаемой маской. Итого, сеть состоит из 23 слоёв.

Принцип обучения заключается в использовании метода стохастического градиентного спуска. Из-за использования свёрток, результирующее изображение меньше исходного на границу ширины. Попиксельное выполнение soft-max функции даёт значение энергии по конечной карте свойств вместе с функцией кросс-энтропии (1). После этого производится расчёт весов (2)

$$E = \sum_x \omega(x) \log(p_{i(x)}(x)) \quad (1)$$

$$\omega(x) = \omega_c(x) + \omega_0 * \exp\left(-\frac{(d_1(x) + d_2(x))^2}{2\sigma^2}\right) \quad (2)$$

Также рассчитывается коэффициент Сёренсена и Жаккара. Основная идея заключается в показателе меры сходства – мера площади правильно выделенных сегментов (S пересечения к S объединения)[4].

Для ускорения процесса работы и повышения производительности используется маска с ограничением области интереса (FOV).

Для оптимального распределения ресурсов, будет использоваться две нейронные сети. Задача первой – сегментация рентгенограммы с удалением шумов и артефактов. Задача второй – непосредственно выявление аномалий легких. Для выполнения второй задачи подготовлена тестовая выборка с заранее определенным классификатором наличия или отсутствия патологий. Данный подход может решить проблему неправильной классификации из-за патологий скелета и мышечного каркаса.

Технология U-NET применялась для решения задачи сегментации клеток глиобластомы с оптического микроскопа и клеток HeLa. Результаты оценивались по критерию IOU (пересечение по объединению) и достигли 92% и 77%, процентов соответственно, что значительно лучше, по сравнению с другими подходами [3].

Name	PhC-U373	DIC-HeLa
IMCB-SG (2014)	0.2669	0.2935
KTH-SE (2014)	0.7953	0.4607
HOUS-US (2014)	0.5323	-
second-best 2015	0.83	0.46
u-net (2015)	0.9203	0.7756

Рисунок 2. Результаты сегментации на соревновании ISBI 2015 г.

Заключение

В статье исследуется процесс сегментации рентгенограмм легких с использованием свёрточной нейронной сети U-Net. Определена структура нейронной сети и основные расчётные процессы. При исследовании предметной области выделены ряд трудностей связанные с сегментацией неточных, искаженных снимков, конкретно – деформация легочного корсета (сколиоз, кифоз, лордоз) приводит к естественным смещениям, но без наличия патологий дыхательных органов, система в таком случае может некорректно выделять область интереса.

Литература

1. High-Throughput Classification of Radiographs Using Deep Convolutional Neural Networks / Alvin Rajkomar, Sneha Lingam, Andrew G. Taylor, Michael Blum, and John Mongan / *Journal of Digital Imaging*. — Feb 2017 — P. 30 - 95.
2. Deep Learning for abnormality detection in Chest X-Ray images / Darwin Yi, Christine Tataru, Archana Shenoyas, Anthony Ma — 2017. — Access mode: <http://cs231n.stanford.edu/reports/2017/pdfs/527.pdf>
3. U-Net: Convolutional Networks for Biomedical Image Segmentation / Ronneberger O., Fischer P., Brox T // *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, Springer, LNCS. — 2015. — Vol. 9351. — P. 234–241.
4. Liskowski et al., "Segmenting Retinal Blood Vessels with Deep Neural Networks", *IEEE Transactions on Medical Imaging*, vol. PP, no. 99, pp. 1-1, 2016.

СЕКЦИЯ 5

МАТЕМАТИЧЕСКОЕ И
КОМПЬЮТЕРНОЕ
МОДЕЛИРОВАНИЕ

ИУС
МКМ 2019

УДК: 004.94

РАЗРАБОТКА ИМИТАЦИОННОЙ МОДЕЛИ КОЛЛ-ЦЕНТРА В СРЕДЕ ANYLOGIC

Нелюбина Ю.А., Харитонов Ю.Е.

кафедра экономической кибернетики

ГОУВПО «Донецкий национальный технический университет»

кафедра экономической кибернетики

E-mail: julia-nelubina@mail.ru

Аннотация:

Нелюбина Ю.А., Харитонов Ю.Е. Разработка имитационной модели колл-центра в среде AnyLogic. В статье рассматривается имитационная модель колл-центра, построенная с использованием имитационной системы Anylogic, в которой, в зависимости от изменения начальных параметров, возможны различные сценарии. Данные отображаются динамически и показывают основные статистические показатели работы колл-центра.

Annotation:

Nelyubina Yu.A., Kharitonov Yu.E. Development of a simulation model of a call center in AnyLogic. The article provides for the Anylogic simulation model, in which various scenarios are possible depending on changes in the initial parameters. The statistical performance of the call center.

Общая постановка проблемы

Для достижения максимального качества обслуживания своих клиентов многие компании готовы вкладывать огромные средства в системы приема и обработки вызовов, так как от этого напрямую зависит количество удовлетворенных обслуживанием клиентов, которые при повторных обращениях влияют на финансовое благополучие и развитие компании.

Исследование процесса функционирования распределенного колл-центра с использованием методов аналитического и имитационного моделирования, а также алгоритмов распределения вызовов в системе колл-центров, необходимое при проектировании и внедрении систем, влияющих на качество обслуживания клиентов, является актуальной задачей.

Исследование

Для моделирования процессов, протекающих в колл-центре, необходимо рассматривать их как последовательность отдельных важных моментов – событий. Для этих целей подходит дискретно-событийное моделирование. Однако каждый пользователь центра индивидуален, и действия разных людей могут отличаться, поэтому для построения модели колл-центра мы будем использовать как агентное, так и дискретно-событийное моделирование.

Модель колл-центра была разрабатываться в среде AnyLogic – инструменте, который предусматривает возможность многоподходного имитационного моделирования.

В данной статье колл-центр рассматривается в качестве двуканальной СМО с ограниченным временем ожидания и нетерпеливыми пользователями. Время ожидания пользователей в очереди становится основным фактором при анализе качества обслуживания. В научных работах данная модель обычно называется моделью Эрланга-А и обозначается как М/М/п+М. Концептуальная модель колл-центра представлена на рисунке 1.

Модель центра обработки вызовов (колл-центра) включает в себя два типа операторов с различным уровнем квалификации и два типа входящих звонков. Модель акцентирует внимание на маршрутизации вызовов. Вызовы поступают с определенными интенсивностями и помещаются в очереди (по одной очереди на каждый тип вызова). Некоторые абоненты могут не дождаться ответа и по истечении определенного времени ожидания покинуть очередь.

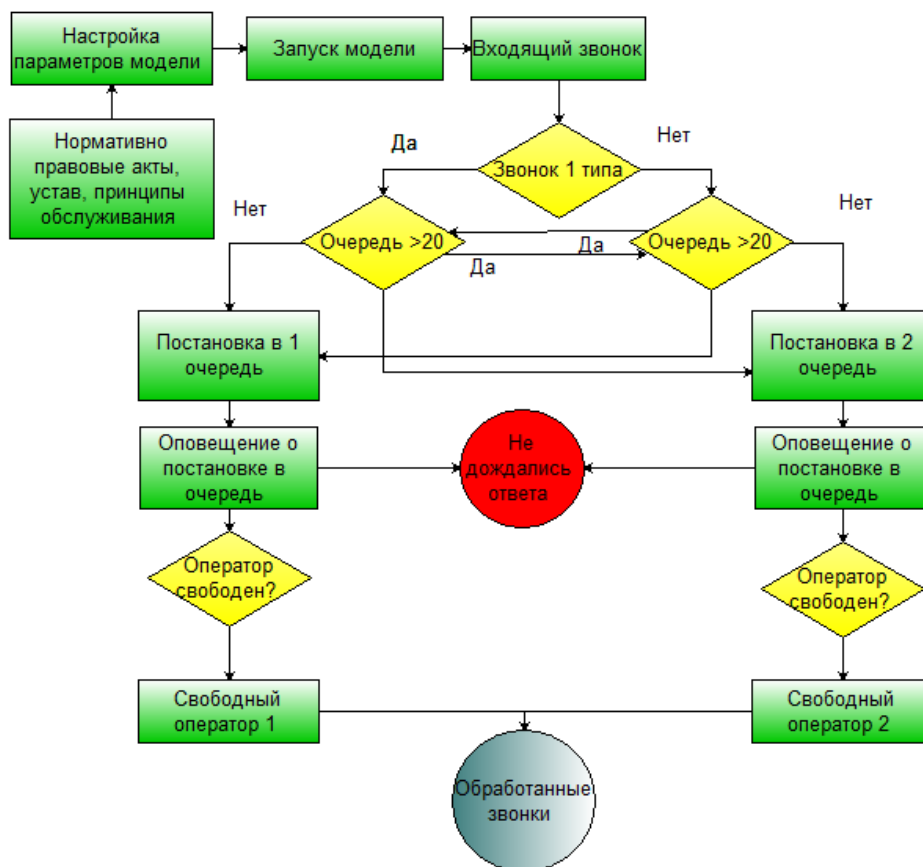


Рис. 1. Концептуальная схема процесса моделирования работы колл-центра

Каждая группа операторов обладает навыками для обработки конкретного типа вызова. Но в то же время они могут обслуживать и непрофильные вызовы, но уже с меньшей эффективностью. Логика маршрутизации вызовов следующая: вызов направляется профильному оператору (если есть свободные операторы этого типа), в противном случае вызов направляется оператору другого типа, опять же, если есть свободный оператор уже этого типа.

В качестве результата модель выдает длины очередей и уровень обслуживания для обоих типов вызовов. Все параметры, в том числе и настройки маршрутизации, могут изменяться по ходу выполнения модели. Параметры модели представлены на рисунке 2. Процессная диаграмма представлена на рисунке 3.

Объект Queue настраивается таким образом, чтобы он позволял агентам покидать объект в том случае, если они ждали в очереди в течение заданного таймаута. Для этого нужно выбрать опцию «Разрешить уход по таймауту» и задать значение таймаута, которое динамически вычисляется для каждого агента. Агенты, покидающие объект по таймауту, направляются в верхний правый порт объекта. Также данные из объекта считываются в переменные для отображения статистики.

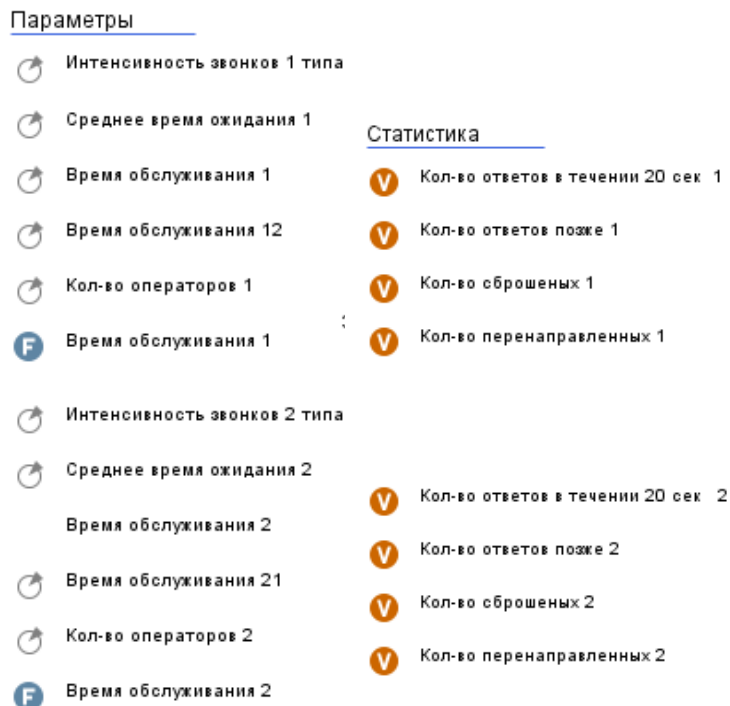


Рис. 2. Параметры и переменные, использованные в модели

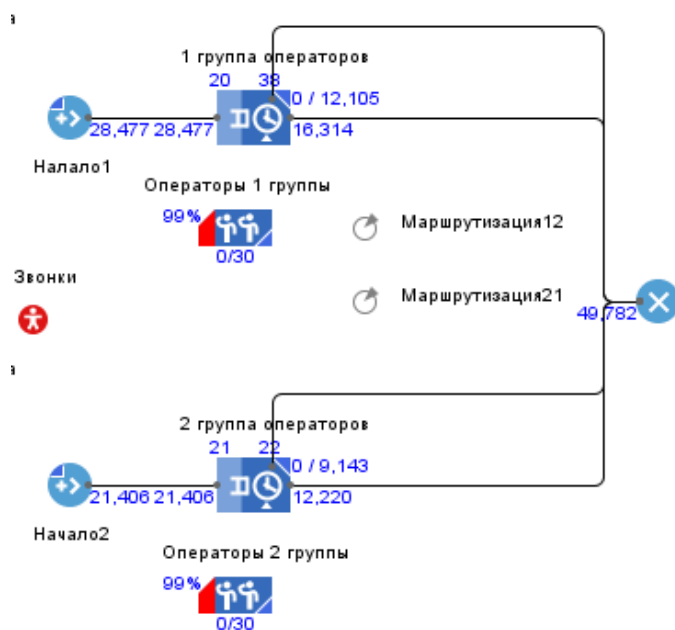


Рис. 3. Процессная диаграмма работы колл-центра

Однако такой вид диаграммы не демонстрирует наглядно процесс приема звонков операторами. Для удобства восприятия информации была создана анимационная форма, наглядно демонстрирующая поток звонков (рисунок 4). Форма позволяет наглядно посмотреть поток звонков и их распределение по операторским линиям, сразу же выводится статистика ожидания в очереди, количество звонков, время ожидания которых менее двадцати секунд, количество потерянных звонков.

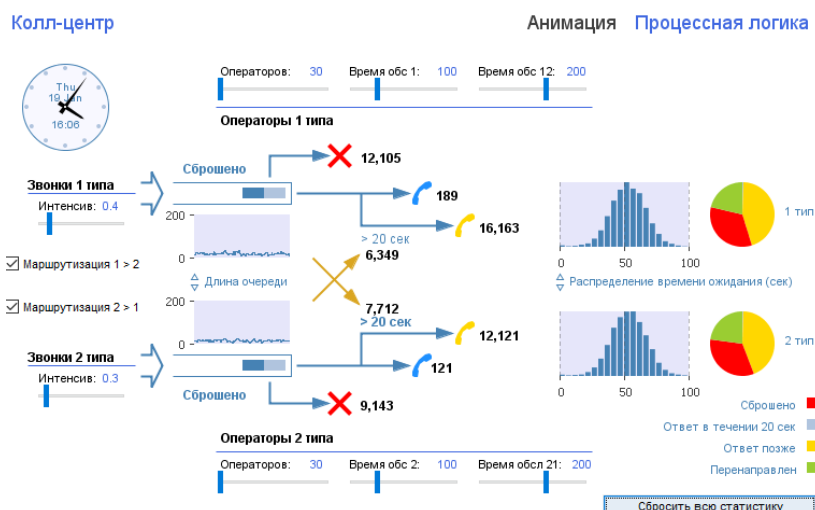


Рис. 4. Анимационная форма

Для управления параметрами модели предусмотрены бегунки. Изменить параметры можно как во время работы модели, так и до ее запуска.

Статистика запросов модели позволит оценить процентное соотношение обработки запросов в зависимости от стадии обработки:

- количество успешно обработанных и потерянных вызовов;
- среднее время ожидания пользователей в очереди;
- среднее время обработки одного вызова;
- количество перенаправленных вызовов.

Благодаря выводу статистики пользователь может оценить работу колл-центра при заданных параметрах. С помощью бегунков параметры можно менять, как перед запуском модели, так и в процессе ее работы. Изменяя параметры модели, пользователь может быстро прийти к необходимому ему варианту.

Выводы

Разработанная модель принимает два типа вызовов с разной интенсивностью, имеет две группы операторов, по одной для каждого типа вызовов. Операторы каждой из групп могут принимать звонки не своего типа, однако время обработки таких вызовов значительно больше. Пользователь может включить или отключить маршрутизацию вызовов, настроить количество операторов каждой группы, интенсивность вызовов, время обработки каждого типа вызовов.

Модель прошла проверку на качество: модель адекватна и соответствует реальному объекту, согласовывается с требованиями и концептуальной схемой. Разработанная модель колл-центра может использоваться для анализа работы отдела и его оптимизации.

Литература

1. Вендров, А. М. Методы и средства моделирования бизнес-процессов (обзор) [Текст] / А. М. Вендров // Информационный бюллетень. – 2004. – № 10 (137). – 32 с.
2. Киселева, М. В. Имитационное моделирование систем в среде AnyLogic [Текст] / М. В. Киселева. – Екатеринбург : УГТУ-УПИ, 2009. – 88 с.
3. Росляков А.В. Ваняшин С.В. Математические модели центров обслуживания вызовов – М.: Ириас. 2006г.

СЕКЦИЯ 6

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
СИСТЕМ ИСКУССТВЕННОГО
ИНТЕЛЛЕКТА

ИУС
МКМ 2019

УДК 004.912

АНАЛИЗ ПРИНЦИПОВ РАБОТЫ ЛИНГВИСТИЧЕСКОГО ПРОЦЕССОРА В ОБРАБОТКЕ ЕСТЕСТВЕННЫХ ЯЗЫКОВ

Михневич П.Д.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа
E-mail: mihnevich.pavel.d@gmail.com

Аннотация:

Михневич П.Д. Анализ принципов работы лингвистического процессора в обработке естественных языков. Рассмотрена работа лингвистического процессора и его предпосылки. Определены задачи выполняемые в каждом блоке работы ЛП. Определены входные и выходные данные, и варианты их использования.

Annotation:

Mihnevich Pavel. Analysis of the principles of the linguistic processor in the processing of natural languages. Consider working language processor and its prerequisites. The tasks performed in each unit of work of the LP are defined. Defined input and output data, and options for their use.

Общая постановка проблемы

Задача обработки текстов на естественном языке появилась в конце 60х годов прошлого столетия. Основной целью отрасли стали системы, способные вести диалог с пользователем на естественном языке. Впоследствии, они разрабатывались долгие десятилетия, и только несколько лет назад достигли относительного успеха. Основным препятствием для создания оных, является тот факт, что ни для одного естественного языка не создана полная и строгая формальная модель. Для полноценного диалога с пользователем, программа должна провести анализ текста, используя знания морфологии, синтаксиса, семантики и прагматики, по которым она определяет «смысл» текста. Для генерации ответа, т.н. «синтеза текста», программе необходимо пройти по всем этим ступеням в обратном порядке. Детальное изучение каждого аспекта задачи в отдельности порождает новые задачи.

Принцип работы лингвистического процессора

Важным элементом фонетического анализа и обработки естественных языков, является лингвистический процессор.

В общем виде, лингвистический процессор состоит из трёх блоков обработки текста: предварительной, пофразовой и пословной.

Первый блок – предварительной обработки текста, принимает на вход орфографический текст. В блоке выполняются следующие операции:

- очистка текста от символов, не имеющих отношения к речи;
- преобразование аббревиатур; (либо в фонетический вид, по звучанию, либо с расшифровкой аббревиатуры);
- перевод чисел в буквенный вид;
- преобразование формул.

Аббревиатуры могут быть преобразованы как в фонетический вид, по звучанию («ДНР» → «дэ эн эр»), либо с расшифровкой («ХозМаг» → «хозяйственный магазин»). Для задач фонетического анализа, в большинстве случаев стоит выполнять преобразование в фонетический вид, т.к. именно в таком виде они используются в речи.

Для последней операции требуется информация об области знания формулы, для её корректного преобразования. Поступающий на выход результат называется нормализованным текстом.

Во втором блоке лингвистического процессора осуществляется пофразовая обработка текста. Получаемый из первого блока нормализованный текст проходит по следующим шагам:

- подвергается членению на фонетические периоды, фразы, синтагмы, в зависимости от ритмической структуры;
- маркировка фразовыми ударениями, разграничивая главные и второстепенные расчленённые части текста;
- интонационная разметка.

Интонационная разметка выполняется в зависимости от знаков препинания в тексте. В конце работы блока, определяется длительность пауз после синтагм. На выходе со второго блока получаем синтагматически размеченный текст.

Последним блоком является блок пословной обработки. Как следует из названия блока, в нём будут выполняться операции над отдельными словами. Этими операциями являются:

- расстановка словесных ударений;
- маркировка ударений конкретным типом ударности (полное, частичное, безударное);
- объединение безударных слов (предлоги, частицы, союзы) с ударными в фонетические слова;
- фонемное транскрибирование.

Суть фонемного транскрибирования заключается в преобразовании отдельных букв в фонемы. Разумеется, одна и та же буква может быть преобразована по-разному, в зависимости от положения в слове, от соседних букв, а также от отдельных исключений.

Заключение

В процессе иерархической обработки текста лингвистическим процессором, текст преобразуется в размеченный фонемный текст, который может быть использован в разных целях. Как пример, для задач синтеза речи, размеченный фонемный текст может быть передан в просодический и фонетический процессор, которые преобразуют его в просодически размеченный фонемный текст и аллофонный текст с акустическими и формантными параметрами текста. Данный набор, пройдя через акустический процессор, будет синтезирован в речевой сигнал.

Литература

1. Бондаренко, Л.В. Фонетическое описание языка и фонологическое описание речи / Л.В. Бондаренко // ЛГУ, 1981. – 200 с.
2. Златоусова, Л.В. Общая и прикладная фонетика / Л.В. Златоусова, Р.К. Потапова, В.В. Потапов // Изд-во МГУ, 1997. – 50 с.
3. Смирнов, Ю.М. Об одном способе построения синтаксического анализатора текстов на естественном языке / Ю. М. Смирнов, А. М. Андреев, Д. В. Березкин, А. В. Брик // Наука, 1997. 42 с.

УДК 004.93

РАЗРАБОТКА МОДУЛЯ СИСТЕМЫ РАСПОЗНАВАНИЯ РУКОПИСНЫХ СИМВОЛОВ

Войдолович И.В., Бычкова Е.В.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: ilya.voidolovich@mail.ru

Аннотация:

Войдолович И.В., Бычкова Е.В. Разработка модуля системы распознавания рукописных символов. Рассмотрены некоторые существующие методы распознавания рукописных символов, подобран подходящий язык реализации и библиотеки методов для создания модуля распознавания.

Annotation:

Voydolovich I.V., Bychkova E.V. Development of handwriting recognition system module. Some existing methods for recognizing handwritten characters are considered, a suitable implementation language and library of methods for creating a recognition module are selected.

Общая постановка проблемы

Перевод бумажных архивов в электронную форму интенсивно осуществляется на предприятиях во всем мире. Зачастую это приводит к непомерным затратам человеческих ресурсов, связанных со слабой степенью автоматизации данного процесса. Проблема заключается в отсутствии эффективной системы, не требующей больших вычислительных мощностей, избавленной от недостатков, связанных с вовлечением человека в процесс распознавания.

Таким образом, возникает необходимость разработки системы распознавания рукописных символов, которая не требует значительных затрат ресурсов компьютера. Для реализации модуля распознавания подобной системы необходимо решить следующие задачи:

- рассмотреть существующие методы, обеспечивающие наибольшую быстроту и точность распознавания при малых затратах машинного времени;
- подобрать подходящий язык реализации и библиотеки методов для создания модуля распознавания.

Цель статьи

Разработать модуль распознавания рукописных символов для минимизации времени на перевод бумажных архивов в электронную форму и освобождения персонала от рутинного труда, связанного с большой вовлеченностью человеческих ресурсов в процесс распознавания.

Актуальность данной статьи

В настоящее время практически все государственные учреждения сталкиваются с проблемами сохранения огромного количества бумажных документов и организации оперативного доступа к содержащейся в них информации. Традиционно эти проблемы решаются с помощью бумажных архивов. Однако это не отвечает требованиям времени: бумажные носители недолговечны, подвержены старению, могут быть искажены, утеряны или уничтожены. Все это может привести к невосполнимой утрате информации. Кроме того, структура расположения бумажных носителей в архиве предполагает использование только одного параметра при поиске необходимого документа (например, по годам или по

алфавиту), что приводит к лишним тратам и потерям времени при получении необходимой информации.

На сегодняшний день единственной альтернативой большому количеству бумаг и документов, находящихся по тем или иным причинам на обязательном хранении, являются электронные архивы. Для того чтоб найти нужную бумагу в обычной картотеке документов, специалист затрачивает в среднем 16 - 18 минут. Для того чтобы найти этот же документ в электронном виде, специалисту понадобится меньше 30 секунд. В связи с этим необходима эффективная система распознавания рукописных символов для перевода бумажных носителей в цифровой вид в промышленных масштабах.

Выбор метода

Распознавание образов – это раздел информатики и смежных дисциплин, развивающий основы и методы классификации и идентификации предметов, явлений, процессов, сигналов, ситуаций и т. п. объектов, которые характеризуются конечным набором некоторых свойств и признаков.

Зачастую для распознавания рукописных символов используют нейронные сети, корректную работу которых обеспечивает большая обучающая выборка объектов. Подобный подход обеспечивает точность распознавания практически до 99,7%.

Однако для решения задачи распознавания возможно использовать и менее тривиальный метод, не требующий больших вычислительных мощностей. В данной работе рассмотрен метод распознавания по затемнению пикселей.

Выбор алгоритма машинного обучения

Не модифицируя метод можно добиться до 50% точности результата, однако получение более высокой точности может обеспечить использование алгоритмов машинного обучения.

Для решения подобной задачи подходит один из самых известных алгоритмов – метод опорных векторов, или Support Vector Machine (SVM).

SVM – набор схожих алгоритмов обучения с учителем, использующихся для задач классификации и регрессионного анализа. Алгоритм принадлежит семейству линейных классификаторов и может также рассматриваться как специальный случай регуляризации по Тихонову. Особым свойством метода опорных векторов является непрерывное уменьшение эмпирической ошибки классификации и увеличение зазора, поэтому метод также известен как метод классификатора с максимальным зазором.

Основная идея метода – перевод исходных векторов в пространство более высокой размерности и поиск разделяющей гиперплоскости с максимальным зазором в этом пространстве. Две параллельные гиперплоскости строятся по обеим сторонам гиперплоскости, разделяющей классы.

В качестве разделяющей гиперплоскости используется гиперплоскость, максимизирующая расстояние до двух параллельных гиперплоскостей. Алгоритм работает в предположении, что чем больше разница, или расстояние, между этими параллельными гиперплоскостями, тем меньше будет средняя ошибка классификатора.

Выбор программных средств

Для программной реализации был выбран высокоуровневый язык программирования Python. Данный выбор был сделан в связи со спецификой формата, в котором находится набор тестовых данных, так как Python позволяет легко загружать наборы данных и манипулировать ими.

Для упрощения реализации используется библиотека Python под названием scikit-learn, которая предоставляет простой интерфейс Python для быстрой библиотеки на основе языка C для SVM, известной как LIBSVM.

Метод опорных векторов в scikit-learn поддерживает оба частых (`numpy.ndarray` и `numpy.asarray`) и редкий (любой `scipy.sparse`) векторы выборки в качестве входных данных. Для оптимальной производительности рекомендуется использовать C-упорядоченный вектор `numpy.ndarray` с типом данных `float64`.

Программная реализация

С учетом вышеперечисленных модификаций метода и рекомендаций к выбору программных средств был реализован модуль распознавания изображения по его среднему затемнению.

Обучение происходило на выборке отсканированных рукописных символов. Обучающая выборка является набором изображений 6000 рукописных символов от «а» до «я» и 1000 тестовых изображений, каждое из которых размером 28x28 пикселей разных оттенков серого цвета. Каждой категории символов в выборке было представлено примерно одинаковое количество.

После запуска классификатора SVM scikit-learn с настройками по умолчанию было правильно распознано 944 из 1000 тестовых изображений, что соответствует 94,4% точности распознавания. Это значительное улучшение по сравнению с начальным подходом к классификации изображения (без модификации метода). Подобные результаты означают, что исследуемый метод уступает в точности нейронным сетям всего на 5,3%.

Результат 94,4% предназначен для стандартных настроек scikit-learn для SVM. SVM имеет ряд настраиваемых параметров, которые улучшают эту готовую производительность. При некоторой работе по оптимизации параметров SVM можно повысить производительность выше 98,5% точности. Другими словами, «хорошо» настроенный SVM будет «ошибаться» только на одной букве из 70.

Выводы

Подобран метод распознавания и наиболее подходящий метод машинного обучения. Для программной реализации выбран высокоуровневый язык программирования и оптимально подходящая библиотека методов. Разработан основной модуль интеллектуальной системы распознавания изображения по его среднему затемнению.

Литература

1. Вьюгин, В.Н. Математические основы теории машинного обучения и прогнозирования / В.Н. Вьюгин– К.: МЦМНО, 2013. – 390 с.
2. Лорьер, Ж.-Л. Системы искусственного интеллекта / Ж.-Л. Лорьер; [пер. с франц.]. – М.: Мир, 1991. – 568 с.
3. Ван Дейк, Т.А. Стратегия понимания связного текста / Т.А. Ван Дейк, В. Кинч ; [пер. с англ.]. // Новое в зарубежной лингвистике. – Вып. 23 "Когнитивные аспекты языка". – М.: Прогресс, 1988. – 211 с.
4. Рашид, Т. Создай свою нейронную сеть / Т. Рашид – MIT Press, 2017. – 228 с.
5. Нелло Кристианини, Джон Шоу-Тейлор. Введение в метод опорных векторов и других методов обучения. - Издательство Кембриджского университета, 2000. – 322 с.

УДК 004.048

ОБЗОР АЛГОРИТМОВ ПОСТРОЕНИЯ КРАТЧАЙШЕГО ПУТИ

Горлов Е.Д., Березовская Е.А., Миненко А.С., Едемская Е.Н.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: zhenya.gorlov@mail.ru

Аннотация:

Горлов Е.Д., Березовская Е.А., Миненко А.С., Едемская Е.Н. Обзор алгоритмов построения кратчайшего пути. Рассмотрена проблема построения кратчайшего пути в рамках улучшения качества обслуживания пассажиров автотранспортных предприятий.

Annotation:

Gorlov E.D., Berezovskaya E.A., Minenko A.S., Edemskaya E.N. Overview of algorithms for constructing the shortest path. The problem of constructing the shortest path in the framework of improving the quality of passenger service for road transport enterprises is considered.

Введение. Надежный контроль движения транспорта является одним из важных условий эффективной логистики на предприятии. Без осуществления постоянного контроля перемещения и текущего местоположения транспорта предприятие не может точно оценивать фактическую ситуацию, предотвращать возникающие проблемные ситуации и оперативно вносить изменения в планы перевозок пассажиров или продукции.

В настоящее время наиболее эффективным и перспективным методом осуществления контроля движения транспортных средств предприятия является использование теории графов и GPS-мониторинга.

Важным фактором успешного учета движения автотранспорта является использование программного обеспечения, способного эффективно анализировать принимаемые данные о местоположении транспортных средств, находить оптимальный маршрут передвижения транспорта и передавать пользователю наглядное изображение текущей ситуации. С развитием GPS-навигации, Интернет-технологий и в связи с дальнейшим внедрением информационных технологий в различных областях хозяйствования, тема мониторинга движения маршрутных транспортных средств автотранспортного предприятия с возможностью отображения на сайте компании представляется необходимой и актуальной.

Общая постановка задачи

Графом называется несколько точек (вершин), некоторые пары которых соединены отрезками (ребрами). Граф связный, если от каждой вершины можно дойти до любой другой по этим отрезкам. Циклом назовем какой-то путь по ребрам графа, начинающегося и заканчивающегося в одной и той же вершине. И еще граф называется взвешенным, если каждому ребру соответствует какое-то число (вес). Не может быть двух ребер, соединяющих одни и те же вершины.

Каждый из алгоритмов будет решать какую-то задачу о кратчайших путях на взвешенном связном. Кратчайший путь из одной вершины в другую – это такой путь по ребрам, что сумма весов ребер, по которым прошли, будет минимальна.

В рассматриваемой предметной области есть несколько городов и дорог, соединяющих эти города. При этом у каждой дороги есть длина. Необходимо попасть из одного города в другой, проехав как можно меньший путь.

Считаем, что в графе n вершин и m ребер.

Алгоритм Флойда-Уоршелла

Находит расстояние от каждой вершины до каждой за количество операций порядка n^3 . Веса могут быть отрицательными, но не может быть циклов с отрицательной суммой весов ребер (иначе можно ходить по нему сколько угодно и каждый раз уменьшать сумму).

В массиве $d[0..n-1][0..n-1]$ на i -ой итерации будем хранить ответ на исходную задачу с ограничением на то, что в качестве «пересадочных» в пути будут использоваться вершины с номером строго меньше $i - 1$ (вершины нумеруются с нуля). Пусть идет i -ая итерация, и необходимо обновить массив до $i + 1$ -ой. Для этого для каждой пары вершин просто попытаемся взять в качестве пересадочной $i - 1$ -ую вершину, и если это улучшает ответ, то так и оставляется. Всего делается $n + 1$ итерация, после ее завершения в качестве «пересадочных» можно использовать любую, и массив d будет являться ответом.

Алгоритм Форда-Беллмана

Находит расстояние от одной вершины (дадим ей номер 0) до всех остальных за количество операций порядка $n * m$. Аналогично предыдущему алгоритму, веса могут быть отрицательными, но не может быть циклов с отрицательной суммой весов ребер.

Заведем массив $d[0..n-1]$, в котором на i -ой итерации будем хранить ответ на исходную задачу с ограничением на то, что в путь должно входить строго меньше i ребер. Если таких путей до вершины j нет, то $d[j] = 2000000000$ (это должна быть какая-то недостижимая константа, «бесконечность»). В самом начале d заполнен 2000000000 . Чтобы обновлять на i -ой итерации массив, надо просто пройти по каждому ребру и попробовать улучшить расстояние до вершин, которые оно соединяет. Кратчайшие пути не содержат циклов, так как все циклы неотрицательны, и мы можем убрать цикл из пути, при этом длина пути не ухудшится (можно также отметить, что именно так можно найти отрицательные циклы в графе: надо сделать еще одну итерацию и посмотреть, не улучшилось ли расстояние до какой-нибудь вершины). Поэтому длина кратчайшего пути не больше $n - 1$, значит, после n -ой итерации d будет ответом на задачу.

Алгоритм Дейкстры

Находит расстояние от одной вершины (дадим ей номер 0) до всех остальных за количество операций порядка n^2 . Все веса неотрицательны.

На каждой итерации какие-то вершины будут помечены, а какие-то нет. Создаются два массива: $mark[0..n-1] - True$, если вершина помечена, $False$ иначе, $d[0..n-1]$ – для каждой вершины будет храниться длина кратчайшего пути, проходящего только по помеченным вершинам в качестве «пересадочных». Также поддерживается инвариант того, что для помеченных вершин длина, указанная в d , и есть ответ. Сначала помечена только вершина 0, а $g[i]$ равно x , если 0 и i соединяет ребро весом x , равно 2000000000 , если их не соединяет ребро, и равно 0, если $i = 0$.

На каждой итерации находится вершина, с наименьшим значением в d среди непомеченных, пусть это вершина v . Тогда значение $d[v]$ является ответом для v . Пусть, кратчайший путь до v из 0 проходит не только по помеченным вершинам в качестве «пересадочных», и при этом он короче $d[v]$. Возьмем первую встретившуюся непомеченную вершину на этом пути, назовем ее u . Длина пройденной части пути (от 0 до u) – $d[u]$. $len \geq d[u]$, где len – длина кратчайшего пути из 0 до v (т. к. отрицательных ребер нет), но по нашему предположению len меньше $d[v]$. Значит, $d[v] > len \geq d[u]$. Но тогда v не подходит под свое описание – у нее не наименьшее значение $d[v]$ среди непомеченных. Противоречие.

Теперь помечаем вершину v и пересчитываем d . Так делаем, пока все вершины не станут помеченными, и d не станет ответом на задачу.

Алгоритм Дейкстры для разреженных графов

Делает то же самое, что и алгоритм Дейкстры, но за количество операций порядка $m * \log(n)$. Следует заметить, что m может быть порядка n^2 , то есть эта вариация алгоритма Дейкстры не всегда быстрее классической, а только при маленьких m .

Что нужно в алгоритме Дейкстры. Нужно уметь находить по значению d минимальную вершину и уметь обновлять значение d в какой-то вершине. В классической реализации пользуются простым массивом, находить минимальную по d вершину возможно за порядка n операций, а обновлять – за 1 операцию. Можно воспользоваться двоичной кучей (во многих объектно-ориентированных языках она встроена). Куча поддерживает операции: добавить в кучу элемент (за порядка $\log(n)$ операций), найти минимальный элемент (за 1 операцию), удалить минимальный элемент (за порядка $\log(n)$ операций), где n – количество элементов в куче.

Создадим массив $d[0 \dots n - 1]$ (его значение то же самое, что и раньше) и кучу q . В куче будем хранить пары из номера вершины v и $d[v]$ (сравниваться пары должны по $d[v]$). Также в куче могут быть фиктивные элементы. Так происходит, потому что значение $d[v]$ обновляется, но нет возможности изменить его в куче. Поэтому в куче могут быть несколько элементов с одинаковым номером вершины, но с разным значением d (но всего вершин в куче будет не более m). Когда берется минимальное значение в куче, надо проверить, является ли этот элемент фиктивным. Для этого достаточно сравнить значение d в куче и реальное его значение. А еще для записи графа вместо двоичного массива используется массив списков.

Волновой алгоритм Ли

Волновой алгоритм – это алгоритм, который позволяет найти минимальный путь в графе. Алгоритм поиска в ширину лежит в основе этого метода. В основном волновой алгоритм применяется для нахождения самого кратчайшего пути в графе, в общем случае находит лишь его длину.

Волновой алгоритм можно назвать одним из самых уникальных алгоритмов трассировки. Волновой алгоритм позволяет сформировать путь (трассу) между двумя ключевыми точками (элементами) в любом лабиринте, здесь необходимо отметить тот факт, что задача является разрешимой.

Исходные данные, цели и задачи, которые требуются для работы волнового алгоритма можно кратко сформулировать следующим образом.

Волновой алгоритм решает задачу нахождения (поиска) пути на плоской двумерной клетчатой карте. Каждой клетке карты присваивается одно из двух состояний "пустая" и "препятствие", также выбираются клетки "начала" и "конца" пути.

Цель волнового алгоритма (как и большинства других алгоритмов) - это задача прокладывания или нахождения пути на карте между начальной и конечной точкой (клеткой).

Волновой алгоритм работает с конца, т.е. из конечной клетки во все направления распространяется волна шагом в одну клетку по радиусу. Далее волна распространяется из соседних клеток и т.д., словно цепная реакция. Этот процесс длится, пока не будет достигнута клетка начала пути или не будут заполнены все поля, т.е. задача не разрешима. Волна движется только по пустым клеткам.

Основная идея волнового алгоритма описывается следующими этапами.

1. Из начального положения (элемента) волна распространяется в 4-х направлениях. Элемент, в который пришла волна, создает новый фронт волны.
2. Каждый из элементов первого фронта волны будет являться источником вторичной волны. Элементы второго фронта волны будут генерировать волну третьего фронта и т.д. Процесс формирования волн продолжается, пока не будет достигнут конечный элемент.

3. На втором этапе волнового алгоритма строится сама трасса. Ее построение необходимо осуществлять в соответствии со следующими правилами:

а) движение при построении трассы необходимо осуществлять в соответствии с выбранными приоритетами;

б) при движении от конечного элемента к начальному, номер фронта волны (путевые координаты) должны уменьшаться.

Приоритеты направления движения при использовании волнового алгоритма нахождения пути выбираются на стадии разработки. Если изменять эти приоритеты, то можно получить разные трассы, но длина трассы в любом случае остается одной и той же.

Выводы. Исходя из достоинств и недостатков рассмотренных алгоритмов, можно сделать вывод, что основное преимущество волнового алгоритма заключается в том, что с его помощью можно найти трассу в любом лабиринте и с любым количеством запретных элементов (стен). Единственным недостатком волнового алгоритма является, то, что при построении трассы требуется большой объем памяти.

Литература

1. Колдаев В.Д. Основы логического проектирования: учеб, пособие. - М.: ФОРУМ: ИНФРА-М, 2011. - 448 с.
2. Соколов А.П. Системы программирования: теория, методы, алгоритмы: учеб, пособие. – М.: Финансы и статистика, 2004.
3. Кнут Д.Э. Искусство программирования. Т. 3. Сортировка и поиск. – 2-е изд. – М.: Вильямс, 2012.
4. Вирт И. Алгоритмы и структуры данных с примерами на Паскале. – СПб.: Невский диалект, 2005.
5. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. - М.: МЦНМО, 2000.
6. Хусаинов Б.С. Структуры и алгоритмы обработки данных. Примеры на языке Си: учеб, пособие. – М.: Финансы и статистика, 2004.
7. Гулаков В.К. Деревья: алгоритмы и программы. – М.: Машиностроение-1, 2005.
8. Мейн М., Савитч У. Структуры данных и другие объекты в C++: пер. с англ. – М.: Вильямс, 2002.
9. Ахо А., Хопкрофт Д., Ульман У. Структуры данных и алгоритмы: учеб, пособие: пер. с англ. – М.: Вильямс, 2000.
10. Мандел Т. Разработка пользовательского интерфейса: пер. с англ. – М.: ДМК Пресс, 2001.
11. Торрес Р. Дж. Практическое руководство по проектированию и разработке пользовательского интерфейса: пер. с англ. – М.: Вильямс, 2002.
12. Бадд Т. Объектно ориентированное программирование в действии: пер. с англ. – СПб.: Питер, 1997.
13. Подбельский В.В. Практикум по программированию на языке Си: учеб, пособие. – М.: Финансы и статистика, 2004.
14. Кубенский А.А. Создание и обработка структур данных в примерах на Java. – СПб.: БХВ-Петербург, 2001.
15. Уайс М.А. Организация структур данных и решение задач на C++: пер. с англ. – М.: ЭКОМ Паблишерз, 2009.

УДК 591.254

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АЛГОРИТМА «FOREL 2» В ФАЙЛОВЫХ СИСТЕМАХ С FAT ТАБЛИЦАМИ

Д.С. Добренко, А.И. Ольшевский

Донецкий национальный технический университет, г. Донецк
кафедра искусственного интеллекта и системного анализа

E-mail: zalman_hack@mail.ru, a_olshevskiy@mail.ru

Аннотация:

Добренко Д.С., Ольшевский А.И. Программная реализация алгоритма «FOREL 2» в файловых системах с FAT таблицами. Рассмотрены современные файловые системы, основанные на таблицах FAT, и выявлены их недостатки. Предложен вариант интеграции алгоритма кластеризации «FOREL 2» для анализа и оптимизации дискового пространства. Определены требования к программной системе и реализован алгоритм «FOREL 2»

Ключевые слова: кластеризация, файловая система, FOREL 2, таблица FAT, дисковое пространство.

Annotation:

Dobrenko D.S. Olshevsky A.I. Software implementation of the FOREL 2 algorithm in file systems with FAT tables. Considered modern file systems based on FAT tables, and identified their shortcomings. A variant of the integration of the FOREL 2 clustering algorithm for the analysis and optimization of disk space has been proposed. The requirements for the software system are defined and the FOREL 2 algorithm is implemented.

Keywords: clustering, file system, FOREL 2, FAT table, disk space.

Файловые системы с FAT таблицами. На сегодняшний день существует большое разнообразие файловых систем, и наиболее распространенная из них – FAT (File Allocation Table). Существует три версии FAT – FAT12, FAT16 и FAT32. Они отличаются разрядностью записей в дисковой структуре, то есть количеством бит, отведённых для хранения номера кластера. FAT12 применяется в основном для дискет, FAT16 – для дисков малого объёма. На основе FAT была разработана новая файловая система exFAT (extended FAT), используемая преимущественно для флеш-накопителей.

Особенностью такой файловой системы является таблица занятости ячеек. Таблица статична и создается 1 раз при формировании нового раздела. Таблица включает в себя ссылки на каждую ячейку раздела, и информацию о том, занята данная ячейка или нет [1-3].

Несмотря на разнообразие существующих файловых систем, во всех них используется в основном один алгоритм записи новых файлов, который основан на последовательном заполнении пустых ячеек данными [1].

Минусы алгоритма:

- чем больше файлов было удалено и перезаписано, тем сильнее становится фрагментация существующих и в дальнейшем записываемых файлов;
- один файл может быть разбит на фрагменты данных, которые будут разбросаны по всему хранилищу данных, что значительно увеличивает время чтения файла;
- размер файла не влияет на выбор места для его записи.

Цель работы – повышение эффективности использования дискового пространства, минимизация фрагментации отдельных файлов, повышение скорости работы с файлом, за счет алгоритма кластеризации «FOREL 2».

Алгоритм кластеризации «FOREL 2»

Суть алгоритма заключается в объединении в кластеры объектов, находящихся в областях наибольшего сгущения [4-5].

На каждом шаге мы случайным образом выбираем объект из выборки, и строим вокруг него сферу радиуса R_0 , внутри этой сферы выбираем центр тяжести и делаем его центром новой сферы. После этого радиус сферы умножаем на значение от 0.6 до 0.9. То есть мы на каждом шаге двигаем сферу в сторону локального сгущения объектов выборки. После того как центр сферы стабилизируется, все объекты внутри сферы с этим центром мы помечаем как кластеризованную и выкидываем их из выборки. Этот процесс мы повторяем до тех пор, пока вся выборка не пройдет кластеризацию.

Программная реализация алгоритма

На первом этапе необходимо проанализировать таблицу FAT и выделить 2 ключевых свойства:

1. Количество, подряд расположенных, свободных ячеек;
2. Расположение, подряд расположенных точек, в таблице FAT.

С помощью данных свойств, сформируем геометрическую плоскость. Осями полученной плоскости будут выступать свойства 1 и 2, что изображено на рисунке 1. На пересечении этих свойств будут объекты, которые и будут проходить кластеризацию.

После анализа необходимо, по свойству 1, отсортировать полученные данные по убыванию. В дальнейшем это поможет быстрее находить оптимальные участки свободного пространства.

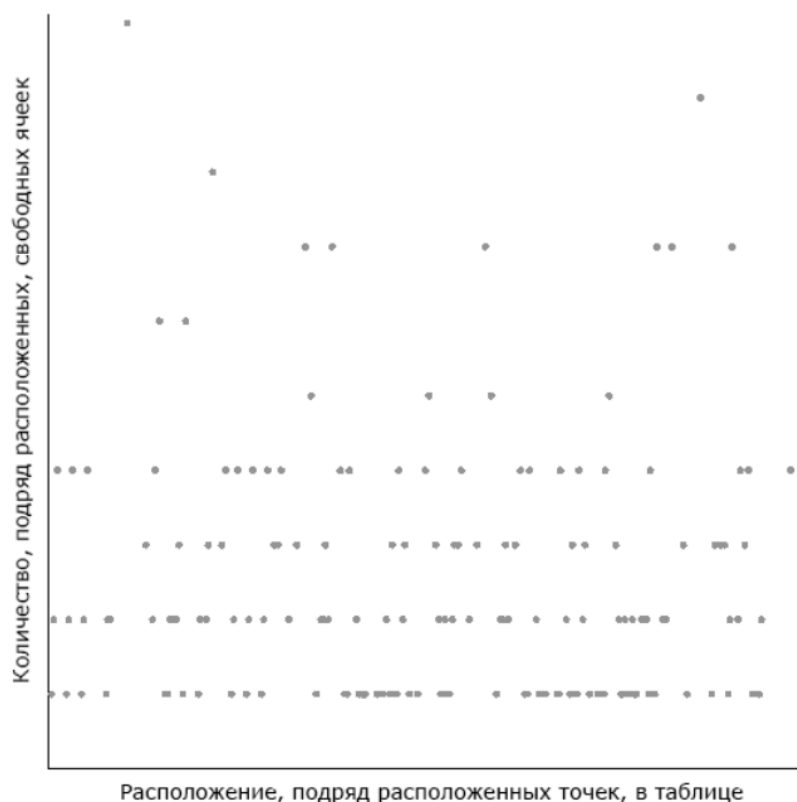


Рисунок 1 – Отображение свойств на геометрической плоскости

Далее, если нам известен размер файла на запись, отсекаем лишние объекты по свойству 1. В качестве верхней планки зададим размер будущего файла, и обозначим его S (см. рис 2).

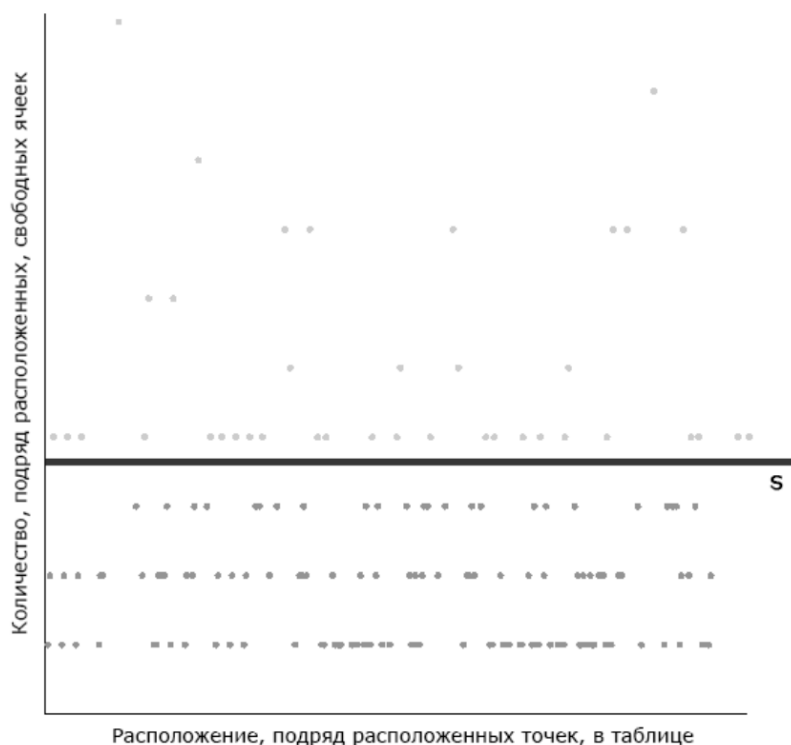


Рисунок 2 – Отсечение объектов

После всех шагов начнем кластеризацию объектов.

1. Строим гиперсферу радиусом R_0 , охватывающую все множество точек;
2. Устанавливаем радиус гиперсферы $R_0 = 0.75$ и перенесем центр сферы в точку с наибольшим значением по оси ординат (количеством свободных ячеек), расстояние до которых меньше радиуса.
3. Вычисляем новый центр тяжести и перенести туда центр сферы
4. Если новый центр тяжести отличается от предыдущего, то возвращаемся к шагу 2 и повторяем цикл. Цикл будет повторяться до тех пор, пока центр тяжести не перестанет смещаться. Таким образом, центр сферы перемещается в область локального сгущения точек.

5. Вычисляем расстояние между центром n -го таксона и всеми точками этого таксона. Если расстояние между центром n -го таксона и точкой k таксона обозначить s_{nk} , то сумма расстояний (P_n) между центром и всеми точками k таксона будет равна:

$$P_n = \sum s_{nk}$$

6. Вычисляем размер свободного пространства n -го таксона и сравниваем его с размером файла (S) на запись. Если размер файла меньше и размер ранее выбранного таксона больше, то сравниваем P_n текущего таксона и ранее выбранного. Если P_n нынешнего таксона меньше, то выбираем его за оптимальный.

7. Точки, принадлежащие текущему таксону, исключаем из анализа, и работа алгоритма повторяется с шага №1. И так до тех пор, пока все точки не будут исключены из анализа.

8. Когда кластеризацию прошли все точки и найден оптимальный таксон, то записываем файл в выделенные таксоном ячейки (см. рис. 3). Если оптимальный таксон не был найден, то увеличиваем значение S на 10% от текущего размера и запускаем кластеризацию заново.

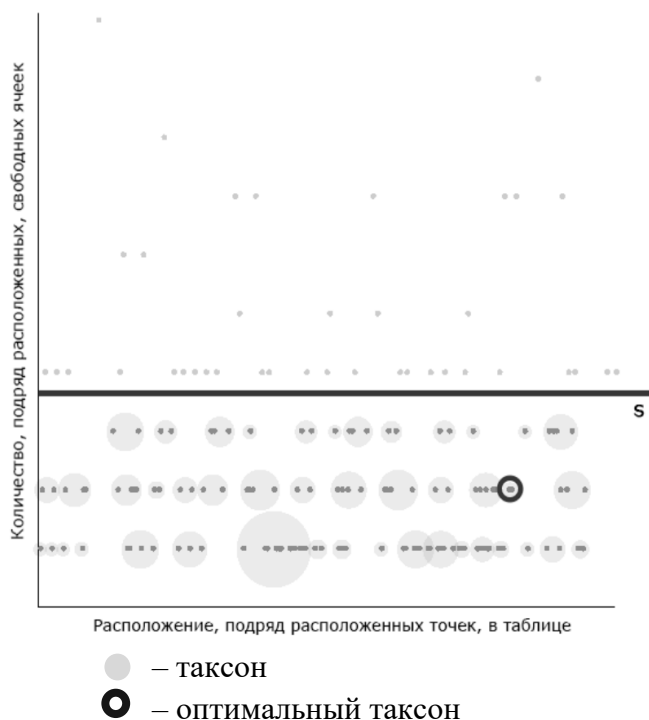


Рисунок 3 – Кластеризация объектов

Выводы

Благодаря использованию модифицированного алгоритма «Форель 2» можно при минимальных мощностях значительно снизить фрагментацию файлов, а также ускорить чтение файлов, исключив большой разброс данных на носителе. Также можно выделить то, что со временем файлы сопоставимых размеров будут распределены на носителе группами, что вносит некую структурированность и открывает новые возможности для создания более быстрых алгоритмов поиска данных и их сортировки.

Литература

13. Таненбаум Э. Современные операционные системы / Э.Таненбаум 3-е изд. – СПб.: Питер, 2010. – 1120 с.: ил.
14. FAT16 и FAT32 [Electronic resource] / Интернет-ресурс. – Режим доступа : <http://www.itstan.ru/komp-tehnika/sravnenie-fajlovyh-sistem-fat16-i-fat32.html>. – Загл. с экрана.
15. Файловая система FAT [Electronic resource] / Интернет-ресурс. – Режим доступа : <http://www.droory.narod.ru/FAT.htm>. – Загл. с экрана.
16. Алгоритмы Форель и Форель 2 [Electronic resource] / Интернет-ресурс. – Режим доступа : <http://www.aiportal.ru/articles/autoclassification/forel.html>. – Загл. с экрана.
17. Эвристический алгоритм Форель [Electronic resource] / Интернет-ресурс. – Режим доступа : <https://studfiles.net/preview/3190827/page:9/>. – Загл. с экрана.

УДК 004.032.26

АНАЛИЗ МОДЕЛИ НЕЙРОНА БОНХОФФЕРА – ВАН ДЕР ПОЛЯ

¹Дробот К.С., ¹Едемская Е.Н., ¹Бычкова Е.В., ²Бельков Д.В.

Донецкий национальный технический университет

¹кафедра искусственного интеллекта и системного анализа, ²кафедра прикладной математики

E-mail: edemskevg@gmail.com, belkovdv@list.ru

Аннотация:

Дробот К.С., Едемская Е.Н., Бычкова Е.В., Бельков Д.В. Анализ модели нейрона Бонхоффера – Ван дер Поля. В работе построена модель нейрона на основе уравнений Бонхоффера – Ван дер Поля. Исследование модели выполнено в среде Octave.

Abstract:

Drobot K.S., Edemskaya E.N., Bychkova E.V., Belkov D.V., The analysis of the neuron model based Bonheoffer Van–der Pol equation. In the paper the model of the neuron is constructing on the base Bonheoffer Van–der Pol equations. The research is executed in the Octave environment.

Общая постановка проблемы

Одним из важных направлений разработки систем искусственного интеллекта является теория нейронных сетей. Искусственные нейронные сети (ИНС) применяют для решения следующих задач:

- классификация объектов;
- кластеризации;
- аппроксимации функций;
- прогнозирования;
- оптимизации;
- создания ассоциативной памяти.

ИНС представляет собой систему взаимодействующих между собой искусственных нейронов. Для эффективной работы она должна использовать модель нейрона, которая является оптимальной при решении задачи. Искусственный нейрон имитирует в первом приближении свойства биологического нейрона. В работе [1] предложена классификация физиологических моделей нейронов, показанная на рис. 1.

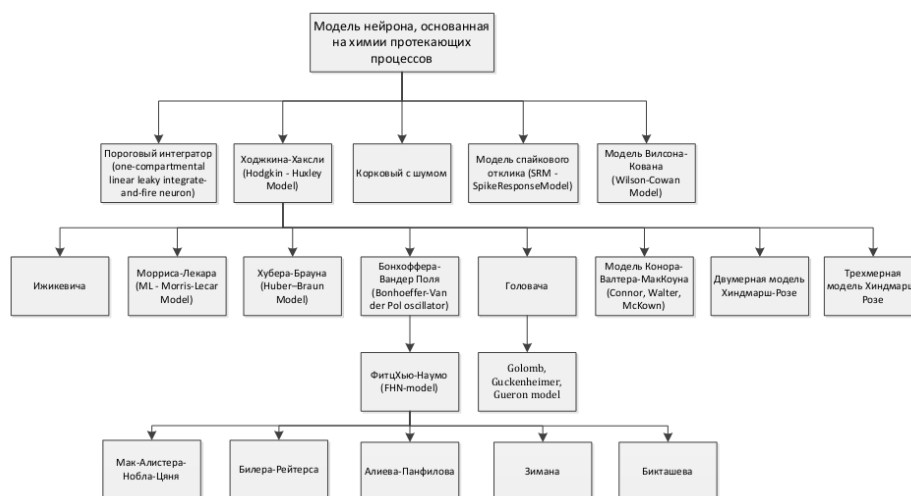


Рисунок 1 - Классификация физиологических моделей нейронов

Одной из физиологических моделей нейронов является модель Бонхоффера – Ван дер Поля. Она была предложена для моделирования нервных импульсов [2,3]. Исследование этой модели – актуальная задача, позволяющая понять, как происходят преобразования временной информации в памяти человека и его органах чувств.

Целью статьи является анализ динамики нейрона. В работе решаются задачи построения и исследования модели Бонхоффера – Ван дер Поля. Показаны результаты вычислительного эксперимента, выполненного в среде Octave.

Исследования

Пусть x – нормированный мембранный потенциал, y – ток активации, ε определяет временной масштаб, параметр a характеризует порог возбудимости. Модель нейрона Бонхоффера – Ван дер Поля имеет вид:

$$\begin{cases} x' = x - y - x^3 / 3 \\ y' = \varepsilon(x + a) \end{cases} \quad (1)$$

Для исследования системы (1) выполнен вычислительный эксперимент в среде Octave при $\varepsilon = 1$. Результаты показаны на рис. 2–7. При $a < 1$ система имеет аттрактор Ван дер Поля. Если $a = 1$, то устанавливается предельный цикл. При $a > 1$ появляется точечный аттрактор.

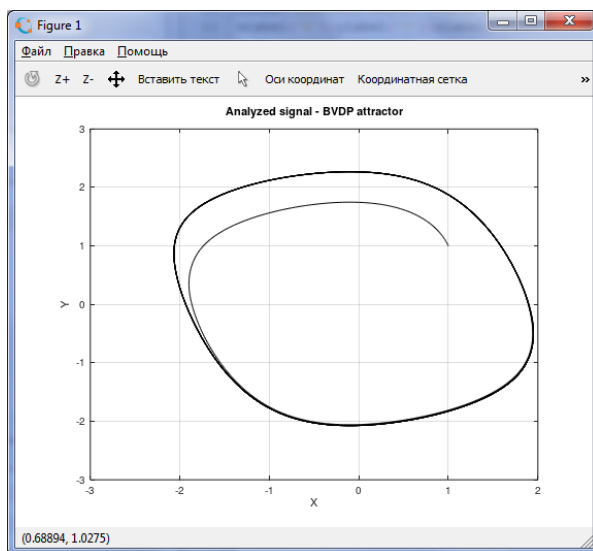


Рисунок 2 – Аттрактор системы при $a=0,1$

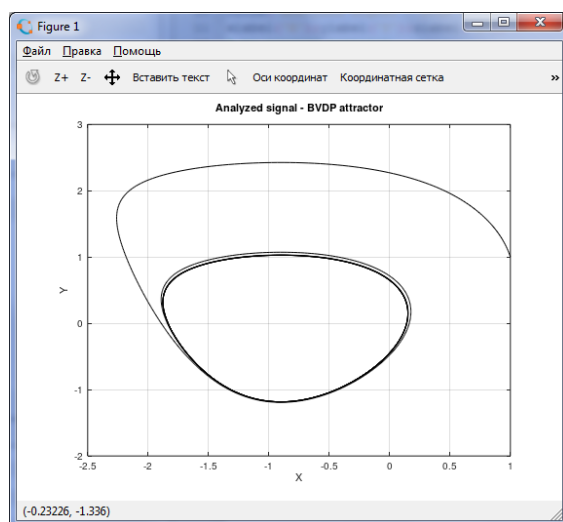


Рисунок 3 – Аттрактор системы при $a=0,9$

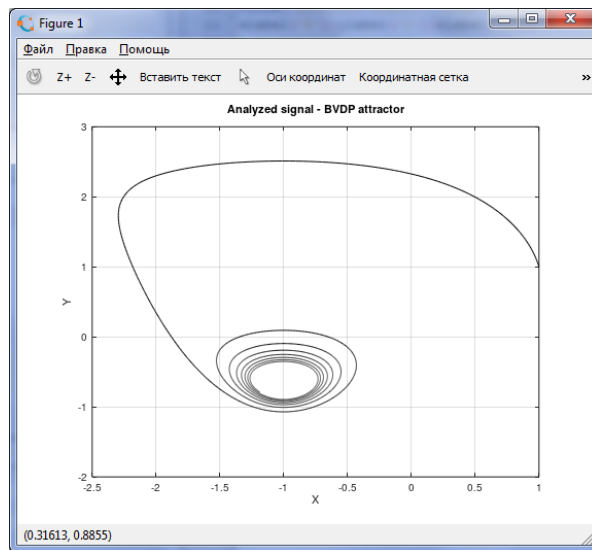


Рисунок 4 – Аттрактор системы при $a=1$

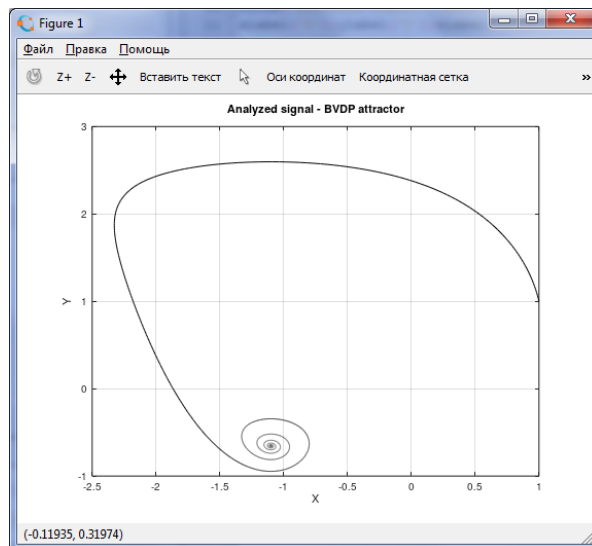


Рисунок 5 – Аттрактор системы при $a=1,1$

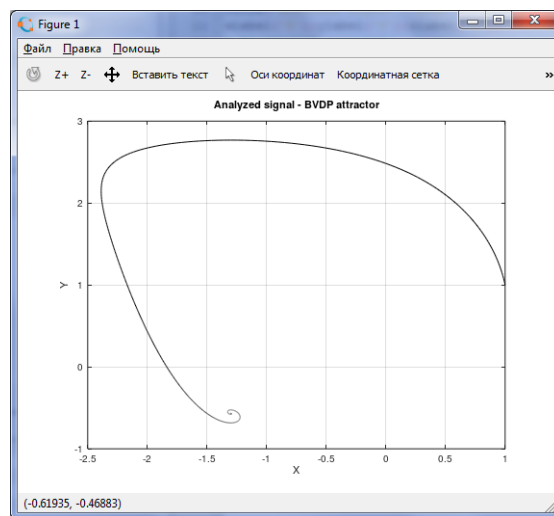


Рисунок 6 – Аттрактор системы при $a=1,3$

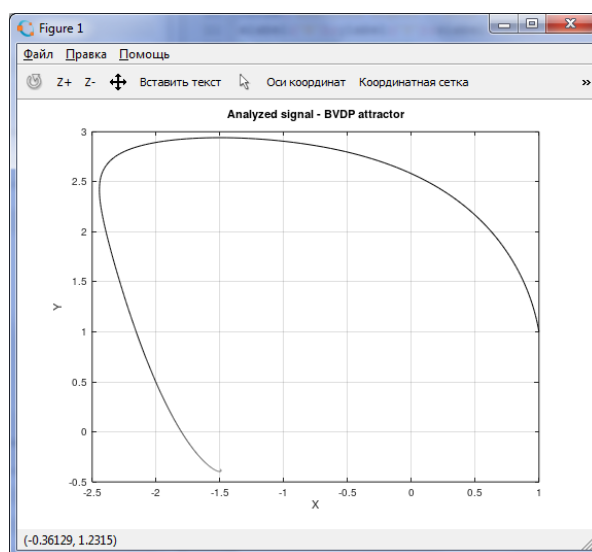


Рисунок 7 – Аттрактор системы при $a=1,5$

Выводы

Искусственные нейронные сети могут быть применены для решения сложных инженерных и академических задач. Основные требования, предъявляемые к искусственным нейронным сетям, – невысокая сложность, быстрая сходимость, устойчивость и надежность полученных результатов. Эти требования во многом обеспечиваются выбором архитектуры вычислителя (искусственного нейрона). Поэтому важной задачей является моделирование искусственных нейронов [4,5].

В работе проанализирована динамика модели нейрона Бонхоффера – Ван дер Поля. Вычислительный эксперимент выполнен в среде Octave. Получены следующие результаты. Если порог возбудимости $a < 1$ система имеет аттрактор Ван дер Поля. При $a = 1$ устанавливается предельный цикл. Если порог возбудимости $a > 1$ появляется точечный аттрактор.

Литература

18. Обзор существующих моделей нейронов. // <https://ailab.ru/investigation/bionika/obzor-sushestvuyushikh-modelei-neironov.html>
19. Осипов Г.В. Синхронизация при обработке и передаче информации в нейронных сетях. Нижний Новгород, 2007, 99 с.
20. Кузнецов А. П., Селиверстова Е. С., Трубецков Д. И., Тюрюкина Л. В. Феномен уравнения Ван дер Поля. // <https://cyberleninka.ru/article/n/fenomen-uravneniya-van-der-polya>.
21. Furber S. Neural Systems Engineering / S. Furber, S. Temple // Studies in Computational Intelligence. – Vol. 115. Computational Intelligence: A Compendium. – Berlin: Springer-Verlag, 2008. – P. 763–796.
22. Нейронные сети в компьютерной инженерии // <http://sdamzavas.net/4-7920.html>

УДК 004.424

ИССЛЕДОВАНИЕ И РАЗРАБОТКА АЛГОРИТМА ДЛЯ РЕКОМЕНДАТЕЛЬНОЙ СИСТЕМЫ

Орлов Ю.К., Жулидов А.Ю.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: ayzhulidov@gmail.com

Аннотация:

Орлов Ю.К., Жулидов А.Ю. Исследование и разработка алгоритма для рекомендательной системы. Определена проблематика задачи. Рассмотрены типы данных необходимых для работы системы. Определены методы получения точных данных. Исследовано влияние временных параметров на работу алгоритма.

Annotation:

Orlov Yu.K., Zhulidov A.Y. Algorithm research and development for a recommender system. The problem is defined. The types of data required for system operation are considered. Defined methods for obtaining accurate data. The effect of temporal parameters on the operation of the algorithm is investigated.

Постановка проблемы

Всемирная сеть сейчас содержит огромное количество информации, знаний. Пользователи на различных условиях могут просматривать всевозможные документы, аудио и видеофайлы. Однако это многообразие данных скрывает в себе проблемы, которые могут возникнуть как при анализе, так и при поиске необходимой информации в сети Интернет.

Это означает, что люди должны ориентироваться среди чрезвычайно большого количества доступных альтернатив, когда хотят что-либо найти. Например, от выбора нового мобильного телефона или плеера до поиска кинофильма для вечернего просмотра. С другой стороны, выступают владельцы интернет-магазинов и сервисов: они заинтересованы в персональной рекламе и рекомендациях каждому конкретному пользователю, потому что такой подход может существенно увеличить прибыль компаний. Как результат, в последние годы интерес к разработке и улучшению существующих рекомендательных систем значительно вырос.

Целью данной работы является исследование данных необходимых для разработки рекомендательной системы, а также анализ используемых для разработки средств и алгоритмов.

Данные необходимые для работы рекомендательной системы

Для работы рекомендательной системы необходимо множество данных. Например, для рекомендательной системы для интернет-магазина, необходима информация о поведении пользователей сайта. Необходимо преобразовать такую информацию в формат, который можно использовать в качестве исходных данных для рекомендательного алгоритма.

Для начала необходимо создать матрицу отношений пользователь-объект. Следующим шагом является обработка явных данных, такие как оценки, которые выставили сами пользователи. И в конце необходимо обрабатывать неявные данные, выяснить что именно является неявными данными, а также разработать способ их использования и вычисления [1].

Рисунок 1 показывает поток обработки данных, которые используются в системе. Сбор данных происходит, когда посетитель пользуется сайтом. Предобработка подразумевает преобразование в данных в формат, подходящий для работы алгоритма. Построение модели и рекомендаций являются заключительными этапами работы системы.

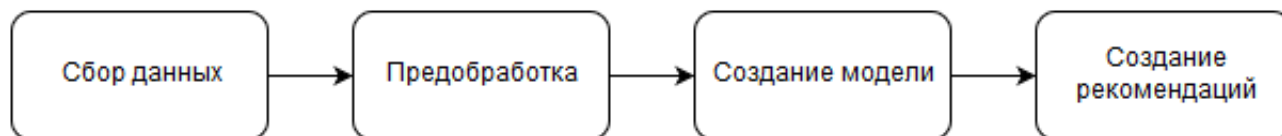


Рисунок 1 – поток обработки данных

Неявные рейтинги должны использоваться потому, что пользователи не уверены в том, что им нравится и даже склонны к тому, чтобы просматривать или делать вещи, которые в других ситуациях сказали бы друзьям (или другим сайтам) что им не нравятся. Например, я посмотрел фильм и рассказал друзьям что фильм мне не понравился, но фильм я всё равно досмотрел. Другая причина использования неявной информации в том, что её намного проще собирать чем явную [2].

Использование надежных источников для рекомендаций

Необходимо определить являются ли ваши пользователи надежными источниками рекомендаций? На некоторых сайтах есть пользователи, продающие вещи, а это значит, что у людей есть стимул выглядеть лучше, а конкурентов – хуже. Например, кому-то пришла отличная идея и он написал книгу об этом. И он был руководителем большой компании, а затем он сказал, что все его сотрудники должны дать положительный отзыв о его книге, иначе их уволят. Тогда страница этого товара на сайте может выглядеть очень хорошо в глазах других пользователей. Или пользователи говорят что-то плохое, потому что товар, который они заказали был сломан, когда его доставили на почту.

Это не значит, что нельзя доверять отзывам на сайте, но нужно думать о том, что побуждает пользователей давать хорошие оценки или плохие оценки. Независимо от того, являются ли они явными или неявными, они всегда могут быть сфальсифицированы [3].

Прочие источники получения данных

Сбор данных о пользователях работает только в том случае, если у вас есть способ уникальной идентификации пользователей. Лучший способ сделать это – сделать так, чтобы пользователь проходил авторизацию, чтобы была идентификация. Другой альтернативой является использование куки.

Обычно сайты начинают с созданием куки и подключения всей информации к этой куки. Если пользователь затем проходит идентификацию, войдя в систему или создав профиль, вся информация из файла cookie передается в эту учетную запись. Необходимо быть осторожными с файлами cookie, потому что компьютер может быть общедоступным или семейным компьютером, которым пользуются несколько человек. Сохранение данных с идентификатором cookie в течение нескольких сеансов может вводить в заблуждение и в итоге служить источником неверных данных.

Если у вас нет авторизованных пользователей, существует также проблема с несколькими устройствами, которая означает, что даже если вы узнаете пользователя на одном устройстве, ваша система не сможет распознать одного и того же пользователя на разных устройствах. Некоторые службы могут помочь вам в этом, но всегда будет неопределенность. Необходимо заставить пользователей зарегистрироваться и войти в

систему, когда это возможно. Само собой разумеется, что персональные рекомендации могут работать, только если вы узнаете пользователя.

Каждый сайт уникален, и данные, которые могут быть собраны на вашем сайте, являются данными, которые лучше всего подходят для выявления поведения ваших пользователей. Иногда нужно получить данные откуда-то еще. Социальные сети – хорошее место для начала, и, если вам повезет, посетителю вашего сайта понравилась информация, соответствующая содержанию в вашем каталоге. В зависимости от вашего содержания, сайтом социальных сетей может быть Facebook, Вконтакте или подобные сайты. Можно подумать, что да, подключение к социальной сети – это хорошо, но если не иметь дело с фильмами или книгами, какова цель добавления данных из социальной сети?

Большинство сайтов, однако, выиграют от получения чего-то столь же простого, как возраст посетителя или место, где они живут. Могут быть и другие способы получения знаний. Рекомендательная система финансовых учреждений могла бы извлечь выгоду из знания, например, читает ли пользователь финансовые книги. Это было бы признаком того, что пользователь заинтересован в финансовой сфере.

Также есть ещё одна вещь которую следует учитывать. Многие алгоритмы рассчитывают рекомендации, основанные на похожих пользователях. Если в системе есть одинаковые данные о многих пользователях, даже если эти данные могут не относиться к содержанию сайта, они всё же позволяют системе находить похожих пользователей, которых затем можно использовать для создания рекомендаций. Например, сайт по продаже автомобилей может обнаружить, что есть особый вид фильма, который нравится большинству владельцев внедорожников, или сайт макияжа, который может порекомендовать что-то по возрасту и полу. Людям, которые работают в IT, вероятно, нравятся гаджеты, а не телефоны, а водителям поездов - солнечные очки.

Подход к рекомендациям на основе времени

Использование бинарной матрицы делает все вещи черно-белыми. Но большинству веб-сайтов нужны нюансы, чтобы система рекомендаций могла лучше понять, что нравится пользователям. В системах рекомендаций самая последняя информация является самой важной. Таким образом, способ сделать матрицу более нюансированной – использовать функцию, основанную на дате покупки. Вы также можете добавить время выхода товара на рынок. Например, покупка, завершённая пять минут назад для товара, который был выпущен (или добавлен в базу данных) пять минут назад, будет иметь более высокий рейтинг, чем покупка пяти минут назад старого товара. У этой покупки также был бы более высокий рейтинг, чем покупка в прошлом году продукта, который был новым в то время.

Выводы

Проведенные исследования позволили выяснить, какие данные необходимо получать для разработки хорошей рекомендательной системы. Установлено, что для рекомендательной системы интернет магазина необходимо большое количество неявных данных, а также обработка данных относительно времени и возможности их фальсификации.

Литература

1. D. Jannach, M. Zanker, A. Felfernig, G. Friedrich Recommender Systems. An Introduction. New York: Cambridge University Press 32 Avenue of the Americas, 2011. 352 P.;
2. P. Melville, V. Sindhvani, Recommender systems. Encyclopedia of Machine Learning, 2010. 169 P.;
3. K. Falk, Practical Recommender Systems. New York: Manning Publications Co. 20 Baldwin Road Shelter Island, 2019. 432 P.

УДК 004

СИСТЕМА УПРАВЛЕНИЯ ВЗАИМООТНОШЕНИЯМИ С КЛИЕНТАМИ

Калужский Г.А., Ольшевский А.И.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа
E-mail: 26gavrik@gmail.com

Аннотация:

Калужский Г.А., Ольшевский А.И. Система управления взаимоотношениями с клиентами. В работе рассмотрены принципы разработки систем управления взаимоотношениями с клиентами. Рассмотрен метод кластеризации Кохонена. Система использует индекс удовлетворенности клиентов и автоматизации обслуживания клиентов.

Annotation:

Kaluzhsky G.A., Olshevsky A.I. Customer Relationship Management System. The article discusses the principles of developing customer relationship management systems. The Kohonen clustering method is considered. The system uses an index of customer satisfaction and automation of customer service.

Общая постановка проблемы

Постоянные клиенты - это жизненная сила любого успешного бизнеса, поэтому для того, чтобы любой бизнес поднялся высоко и завоевал признание, важно тщательно развивать отношения с клиентами. Однако при росте компании сделать это без помощи передовых организационных технологий практически невозможно. Со временем объем данных о клиентах, которые необходимо отслеживать, может оказаться слишком большим для простой электронной таблицы. И в большинстве случаев собранная информация почти не используется для привлечения и удержания клиентов.

Таким образом, возникает необходимость разработки универсальной системы управления взаимоотношениями с клиентами.

Цель статьи

Разработать алгоритм управления взаимоотношениями с клиентами компании для увеличения прибыли, за счет более эффективного управления и создания информационного обеспечения взаимоотношений с клиентами.

Исследование

Компьютерная система управления взаимоотношениями с клиентами – новая форма взаимодействия, заключающаяся в том, что главным в философии торговли будет именно клиент. Система состоит из центральных этапов деятельности, в ней будут применяться все методы, которые направлены на поддержку маркетинга, улучшение продаж и уровня качества эффективной работы с клиентами. Информационный цикл работы системы (рис 1.) должен затрагивать весь жизненный цикл клиента.

Основные задачи CRM заключаются в полном удовлетворении потребителя системой автоматизированного управления за счет качественного анализа полученной информации о клиентах, урегулировании ценовой политики, улучшении инструментов рекламы. Получив эффективную централизованную обработку информации, возникает прекрасная возможность, не привлекая большое количество сотрудников, сотрудничать с несколькими заказчиками одновременно, учитывая индивидуальные нужды каждого из них. Быстрая обработка информации о клиентах дает возможность исключить всевозможные риски и оценить потенциальные успехи любой сделки. Для этого используются методы сегментации.



Рисунок 1 – Цикл информационных процессов в CRM

Метод кластеризации клиентской базы на основе сети Кохонена

Сегмент клиентов – это группа реальных, потенциальных потребителей с однотипными потребностями и поведенческими реакциями на продукт.

Сегментация помогает определить продуктовые, маркетинговые, рискованные и прочие стратегии взаимодействия с определенной группой клиентов. Иначе говоря, решить кому, как и что продавать так, чтобы точно захотели купить.

Сегментация или сегментирование клиентов – это процесс разделения потребительского рынка на части (сегменты) согласно определенным признакам. Для этого идеально подходят самоорганизующиеся карты Кохонена. Это мощный самообучающийся механизм кластеризации, позволяющий отобразить результаты в виде компактных и удобных для интерпретации двумерных карт.

Данный обработчик используется для поиска закономерностей в больших массивах данных. Это позволяет проводить разведочный анализ данных, отличающийся от классических статистических процедур, в ходе которых проверяется некоторый набор выдвинутых гипотез.

Основные преимущества алгоритма:

- устойчивость к зашумленным данным
- быстрое и неуправляемое обучение
- возможность визуализировать многомерные входные данные

Разные группы потребителей требуют различного подхода. Сегментация в CRM помогает сократить время на обработку заявок, укрепить деловые отношения с крупными заказчиками, отсеять нецелевую аудиторию. Это позволяет более качественно работать с целевым сегментом и выявлять степень удовлетворенности клиентов продуктом. Таким образом можно увеличить число благодарных клиентов и увеличить прибыль.

Индекс удовлетворенности потребителей

Индекс удовлетворенности потребителей – маркетинговый показатель и метод оценки качества маркетинга компании.

Цель расчета индекса удовлетворенности потребителей – точнее оценить качество маркетинга компании. Задачи, которые решает оценка удовлетворенности:

- выявить слабые места комплекса маркетинга и скорректировать комплекс;
- выявить ошибки сервис-менеджмента;
- оценить качество менеджмента по обслуживанию потребителей;
- выявить элементы бизнес-модели, которые компании стоит улучшить;

Метод основан на опросе потребителей, совершивших то или иное действие в отношении товара, марки или их владельца. Опрос может осуществляться в магазине после покупки, по телефону, если покупатель оставил координаты, по электронной почте и любыми другими доступными методами опроса.

Автоматизация обслуживания клиентов

Одним из способов удержания клиента является повышение качества обслуживания. Автоматизация этого процесса сокращает время ожидания и улучшает опыт взаимодействия с клиентом.

Функциональность автоматизации обслуживания клиентов включает:

- базу данных контактов с клиентом, возможность групповой работы с клиентом;
- управление взаимоотношениями с потенциальными клиентами, деловыми партнерами;
- интерактивная поддержка клиентов, предоставление сервисных служб;
- мониторинг поступления заявок - контроль процессов обработки запросов и заявок, реакции на них,
- отчетность о результатах обслуживания;
- скидки и бонусные системы, дистанционное предоставление услуг;
- обслуживание клиентов и деловых партнеров в режиме реального времени;
- базу знаний типичных проблем и способов их решения;
- автоматическое отслеживание прекращенных сроков договоров.

Выводы

Рассмотрены основные компоненты систем управления взаимоотношения с клиентами. Представлены метод кластеризации Кохонена. Определены главные направления системы управления взаимоотношения с клиентами. Разработаны основные требования к составлению тестовых заданий для интеллектуальной системы автоматизации контроля уровня знаний.

Литература

23. Гринберг, Пол. CRM со скоростью света / Пол Гринберг. – Спб.: Символ плюс, 2007. – 528 с.
24. Схиртладзе, А.Г. Автоматизация производственных процессов / А.Г. Схиртладзе. – Старый Оскол.: ТНТ (Тонкие наукоемкие технологии), 2013. – 421 с.
25. Новиков А.В. Нейросетевые методы решения задач кластерного анализа / А.В. Новиков, Е.Н. Бендерская // Нейрокомпьютеры: разработка, применение – М.: Радиотехника 2014. – С. 48–53.
26. Система управления взаимоотношениями с клиентами [электронный ресурс] // Википедия – свободная энциклопедия: [сайт]. [2016]. URL: https://ru.wikipedia.org/wiki/Система_управления_взаимоотношениями_с_клиентами
27. Net Promoter Score and System (NPS): an introduction [электронный ресурс] // I-scoop: [сайт]. [2019]. URL: <https://www.i-scoop.eu/customer-experience/net-promoter-score-nps/>

УДК 519.71

О ВЛИЯНИИ ПЕРЕБРОСКИ ДУГ НА ДЕГРАДАЦИЮ ПОВЕДЕНИЯ КОНЕЧНОГО АВТОМАТА

Ковалёв Д.В., Копытова О.М.

Донецкий национальный университет, г. Донецк
кафедра искусственного интеллекта и системного анализа
Email: danilkovalyov2805@gmail.com, omkop@list.ru

Аннотация

Ковалёв Д.В., Копытова О. М. *О влиянии переброски дуг на деградацию поведения конечного автомата.* Исследуется поведение конечного приведенного автомата под действием переброски дуг в графе его переходов. Найдены условия, при которых преобразованный автомат имеет наименьшее возможное число различных состояний, т.е. максимально деградировавшее поведение.

Ключевые слова: автомат, поведение, переброска дуг.

Abstract

Kovalyov D. V., Kopytova O. M. *On the effect of the arcs transfer on the degradation of the behavior of a finite state machine.* The behavior of a finite reduced state machine under the action of arcs transferring in its transitions graph is investigated. The conditions under which the transformed automaton has the smallest possible number of distinguishable states, i.e. maximum degraded behavior, are found.

Keywords: finite state machine, behavior, arcs transfer.

Введение

Конечный автомат – математическая модель, используемая для изучения широкого класса преобразователей дискретной информации, которые могут быть как реальными устройствами или процессами, так и абстрактными системами. Автоматная модель применяется в таких областях, как программирование, моделирование логических устройств, техническая диагностика, теория динамических систем и т.д.

В настоящей работе рассматриваются конечные автоматы, описывающие такие дискретные системы, которые, функционируя во внешней среде, могут подвергаться повреждающим воздействиям со стороны этой среды. Вследствие этого может измениться как структура системы, так и её поведение. Соответствующие изменения в структуре и поведении автомата, моделирующего систему, в ряде случаев можно адекватно описать с помощью специального преобразования его графа переходов, задаваемого перебросками дуг в графе. Если при этом отметки дуг (пары вход-выходных символов) не меняются, то такие переброски можно понимать как "неисправности" функции переходов автомата.

В работах [1-3] исследовалось поведение автоматов, порождаемых друг из друга с помощью переброски дуг. При этом основной задачей было нахождение условий, при которых переброска дуг в автомате приводит к изоморфному автомату, т.е. поведение автомата до и после переброски остается неизменным. Были найдены соответствующие критерии для случаев инициального и неинициального конечного приведенного автомата Мили.

В настоящей работе решается противоположная задача – ищутся такие переброски, которые максимальным образом изменяют поведение исходного приведенного неинициального автомата Мили. При этом под поведением автомата понимается множество

вход-выходных деревьев, порожденных всеми его состояниями, а изменение поведения измеряется количеством появившихся эквивалентных состояний.

Заметим, что в результате любой переброски автомат может либо остаться приведенным, либо же некоторые его состояния могут оказаться эквивалентными. Если после переброски число различных состояний автомата уменьшилось, то будем говорить о деградации поведения автомата. Чем больше появилось эквивалентных состояний, тем соответственно меньше осталось попарно различных, т.е. тем больше деградировало поведение автомата.

Постановка задачи

Основные определения теории автоматов можно найти в [4].

Под автоматом будем понимать приведенный автомат Мили $A = (S, X, Y, \delta, \lambda)$, где S, X, Y – конечные множества состояний, входных и выходных символов соответственно, а δ, λ – функции переходов и выходов. Пусть $n = |S|$.

Два состояния s и t одного и того же автомата A или двух разных автоматов A и B соответственно, называются эквивалентными если для всякого входного слова $p \in X^*$ выполняется $\lambda(s, p) = \lambda'(t, p)$, где λ' – функция выходов автомата A или B . В противном случае s и t называются различными. Автомат называется приведенным, если все его состояния попарно различимы.

Каждому состоянию $s \in S$ автомата A поставим в соответствие множество λ_s всех вход-выходных слов конечной длины, порождаемых этим состоянием.

Автомат в нашем случае удобно задавать в виде графа переходов, вершины которого соответствуют состояниям из S , а дугами являются четверки (s, x, y, t) , где $t = \delta(s, x)$, $y = \lambda(s, x)$. Пара (x, y) называется отметкой дуги, s – её началом, а t – концом. Если $e = (t, x, y, u)$ – дуга в графе переходов автомата A , то переброской этой дуги из состояния u в состояние v , отличное от u , называем замену её дугой (t, x, y, v) . Переброску k дуг для краткости назовём k -переброской.

В работе исследуются такие переброски в автомате-эталоне A , в результате которых получается автомат A' , содержащий эквивалентные состояния. Будем говорить, что некоторая k -переброска приводит к i -деградации автомата, если после этой переброски число попарно различных состояний в A' равно i , где $i < n$. Нас интересуют переброски наименьшего числа дуг, приводящие к максимальной деградации исходного автомата A при тех или иных ограничениях на сами переброски. Максимальная деградация соответствует минимальному числу различных состояний в автомате A' . Целью исследования является нахождение указанных перебросок.

В процессе исследования необходимо решить следующие задачи:

- построить примеры автоматов, когда переброска дуг приводит к эквивалентным состояниям;
- выяснить существование перебросок, приводящих к максимальной деградации автомата, и описать способ их построения.

Результаты исследования

1. Рассмотрим пример приведенного автомата A на рисунке 1.

Переброска дуги $(2, 0, 0, 4)$ в состояние 3, т.е. замена её дугой $(2, 0, 0, 3)$, приводит к автомату A' , в котором состояния 1, 2 и 3 эквивалентны, т.е. приводит к 2-деградации поведения A .

2. Заметим, что количество состояний после любой переброски не может быть меньше, чем число вход-выходных деревьев высоты 1, порождаемых всеми состояниями автомата A . Обозначим это число через l . В нашем случае таких деревьев два, т.е. $l=2$, так как состояния 1, 2 и 3 порождают одно и то же дерево $\{0/0, 1/0\}$, а состояние 4 – дерево $\{0/1, 1/0\}$.

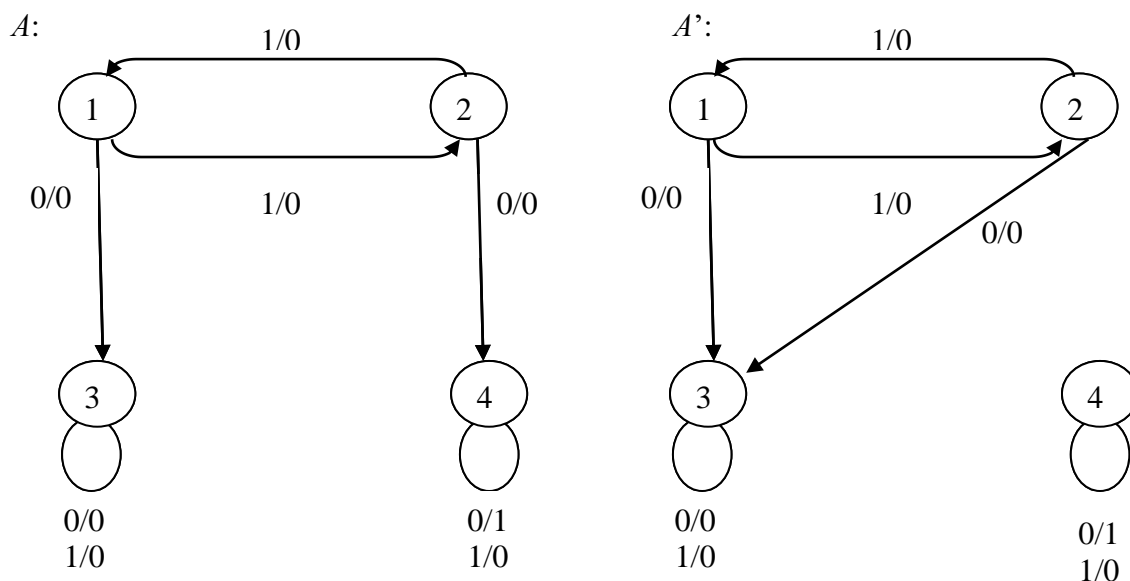


Рисунок 1 – Автомат до и после переброски дуги (2,0,0,4) в состояние 3

Построим таблицу P_1 1-эквивалентных состояний автомата A [5] и посмотрим как она изменится для A' при переброске дуги (2,0,0,4).

		x		
		s	0	1
A	1	3 A	2 A	
	2	4 B	1 A	
	3	3 A	3 A	
B	4	4 B	4 B	

Таблица 1 - P_1 для A

		x		
		s	0	1
A	1	3 A	2 B	
	3	3 A	3 A	
B	2	4 C	1 A	
C	4	4 C	4 C	

Таблица 2 – P_2 для A

		x		
		s	0	1
A	1	3 A	2 A	
	2	3 A	1 A	
	3	3 A	3 A	
B	4	4 B	4 B	

Таблица 3 – P_1 для A'

Из таблицы 1 видно, что состояния 1 и 3, принадлежащие классу A, имеют одинаковые отметки по всем входным символам. Следовательно, в таблице P_2 они окажутся в одном классе, т.е. будут 2-эквивалентны (см. таблицу 2). Состояние 2 будет также находиться в одном классе с 1 и 3, если по нулю перейдет в какое-либо состояние из класса A, а не B, например, в состояние 3 (см. таблицу 3). Ясно, что таблица P_2 для A' совпадет с таблицей P_1 , т.е. состояния 1, 2 и 3 в автомате A' будут эквивалентны.

Таким образом, в результате замены дуги (2,0,0,4) дугой (2,0,0,3) получился автомат A' , в котором три состояния эквивалентны между собой. Следовательно, данная переброска привела к максимальной 2-деградации автомата A , поскольку, как сказано выше, у автомата A порождает два дерева высоты 1.

Рассмотренный пример позволяет описать алгоритм нахождения минимального по мощности множества дуг, переброска которых всегда приводит к максимальной деградации автомата-эталона. Приведем словесное описание этого алгоритма.

1. В таблице P_1 число классов 1-эквивалентности совпадает с l . Пусть j – номер рассматриваемого класса, $1 \leq j \leq l$. Сначала $j=1$.

2. Рассматриваемый j -ый класс по каждому входному символу переходит в некоторое множество состояний-преемников с соответствующими отметками. Обозначим через $C(j,x)$ список отметок состояний-преемников j -го класса по x . Например, $C(1,0)=(A,B,A)$, т.е.

первый класс (класс с отметкой А) по нулю переходит в состояния 3,4,3 с отметками А,В,А соответственно.

3. В этом списке отметка некоторого класса (одного или нескольких), встречается наиболее часто. Выберем одну из таких отметок. В примере это отметка А. Все остальные отметки также превратим в выбранную, заменив соответствующие им состояния на любое состояние из выбранного класса. В примере отметка В была заменена на выбранную отметку А в результате замены состояния-преемника 4 на состояние 3. Для автомата А это означает замену дуги $(2,0,0,4)$ дугой $(2,0,0,3)$, т.е. переброску этой дуги в состояние 3.

4. Описанные действия повторим каждого x текущего класса, после чего перейдем к следующему классу: $j = j + 1$.

5. Когда все классы таблицы P_1 будут рассмотрены, будет получена таблица P_1 , в которой каждый класс 1-эквивалентности по любому x переходит в некоторый класс этой же таблицы. Отсюда следует, что $P_2 = P_1$. А это означает, что построен автомат с максимальной деградацией поведения. Из построения ясно, что количество перебросенных дуг при этом минимально, так как число дуг, оставшихся без изменений, максимально (оно совпадает с числом наиболее часто встречающейся отметки класса среди состояний-преемников).

Выводы

В работе введено понятие деградации поведения автомата в результате переброски дуг в его графе переходов. Определена максимально возможная деградация для заданного автомата-эталона. Описан алгоритм поиска множества дуг минимальной мощности, переброска которых приводит к максимальной деградации. Открытым остается вопрос, какова максимальная деградация, к которой приводит переброска в точности одной дуги.

Литература

1. Копытова О.М. О сравнении поведения автоматов, порождаемых локальными преобразованиями ОД-к-эталона / Труды IX Международной конференции "Дискретные модели в теории управляющих систем" (Москва и Подмосковье, 20–22 мая 2015 г.). М., Макс-Пресс, 2015. – С. 113 – 116.
2. Горяйнова А. В., Копытова О. М. О перебросках дуг, приводящих к изоморфным автоматам // Труды X Международной конференции «Дискретные модели в теории управляющих систем» (Москва и Подмосковье, 23–25 мая 2018 г.). Отв. ред. В. Б. Алексеев, Д. С. Романов, Б. Р. Данилов. – Москва: МАКС Пресс, 2018. – С. 113 – 116.
3. Гербутова Д. А., Копытова О. М. Об устойчивости поведения автоматов-распознавателей к локальным преобразованиям // Материалы студенческой секции IX Международной научно-технической конференции «Информатика, управляющие системы, математическое и компьютерное моделирование» (ИУСМКМ-2018). – Донецк: ДонНТУ, 2018. – С. 309 – 314.
4. Кудрявцев В. Б., Алешин С. В., Подколзин А. С. Введение в теорию автоматов. М. : Наука, 1985. – 320 с.
5. Мур Э.Ф. Умозрительные эксперименты с последовательностными машинами // Автоматы. – М. : Иностран. лит. – 1956. – С. 179–210.

УДК 519.683

ИССЛЕДОВАНИЕ И РАЗРАБОТКА МЕТОДОВ ГЕНЕРАЦИИ ТРЁХМЕРНЫХ ИЗОБРАЖЕНИЙ ПРИ ПОМОЩИ НЕЙРОННЫХ СЕТЕЙ

Кошман А. Д., Ольшевский А.И.

Донецкий национальный технический университет, г. Донецк
кафедра искусственного интеллекта и системного анализа

E-mail: rainbovvdash333@gmail.com, a_olshevskiy@mail.ru

Аннотация

Кошман А.Д., Ольшевский А.И., Исследование и разработка методов генерации трёхмерных изображений при помощи нейронных сетей. В работе рассмотрены основные тенденции развития алгоритмов распознавания изображений при помощи свёрточных нейронных сетей, базовые понятия работы нейронных сетей, алгоритм распознавания жилых домов по фотографическим изображениям.

Ключевые слова: свёрточные нейронные сети, алгоритм распознавания изображений, типы жилых домов.

Abstract

Koshman A.D., Olshevsky A.I. Research and development of methods for generating 3D images using neural networks. The paper considers the main trends of the development of image recognition via convolutional neural networks, the basic concepts of neural networks functioning, algorithm of condominium recognition by photographic images.

Key words: convolutional neural networks, image recognition algorithm, types of condominiums.

Введение

В современном мире у людей появилась возможность посещать различные места за сотни и тысячи километров при помощи виртуальных карт. Такие сервисы, как Google Maps или же Yandex карты предоставляют такие услуги, но, увы, большинство регионов люди могут наблюдать пока только в виде фотографий со спутников [1-3]. Следовательно, существует необходимость в развитии технологий, которые позволили бы взглянуть на местность с точки зрения туриста.

Целью данной работы является исследование существующих алгоритмов анализа изображений посредством машинного зрения, а также выявление их преимуществ и недостатков для генерации трёхмерных изображений.

Задачей исследования являются:

- провести обзор программных продуктов на основе нейронных сетей для работы с графикой;
- разобраться со структурой и принципом работы нейронных сетей;
- выбрать подходящую архитектуру нейронной сети.

Результатом работы должна стать программа, позволяющая создавать упрощенную 3D модель зданий улиц (всего города) на основе электронной базы карт Google.map и имеющихся в этой же базе соответствующих 2D снимков[4].

Исследования

Сегодня широко используют искусственную нейронную сеть (artificial neural network) при решении задач компьютерного зрения. Теоретические основы нейроматематики были заложены в начале 40-х годов. В 1943 году У.Маккалох и его ученик У.Питтс сформулировали основные положения теории деятельности головного мозга. Нейронные

сети построены на технологии работы нейронов в головном мозге [4-7]. Принцип работы нейрона приведён ниже на рисунке 1.

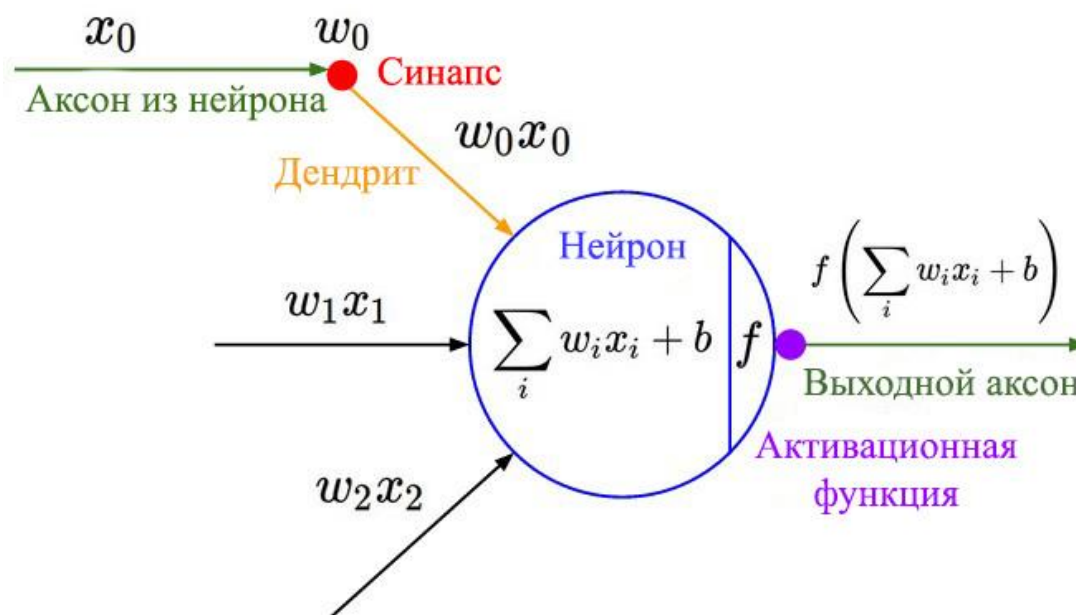


Рисунок 1 – Принцип работы нейрона

На вход в нейрон поступают сигналы от других нейронов через аксоны. Данные о предыдущих нейронах вычисляются по формуле:

$$\sum_i w_i x_i + b \quad (1)$$

где w_i – вес входа нейрона i ;
 x_i – сигнал на вход нейрона i ;
 b – коэффициент смещения нейрона.

Свёрточные нейронные сети могут быть использованы для реализации данной технологии [8-10]. Преимущество свёрточных сетей над другими состоит в принципе их работы: для анализа изображения сеть использует разбиение изображения на свёртки, каждая из которых имеет множество других более мелких свёрток. Самый нижний уровень свёртки сканируется нейронной сетью по направлению слева направо и сверху вниз. Оптимальная размерность квадрата сканирования находится в пределах от 3x3 до 11x11 пикселей. Данный подход даёт вероятность распознавания образов в 80%, что сравнительно отличается от одной из распространённых моделей нейронных сетей «Перцептрон», чьи результаты составляют 50%.

Для генерации трёхмерных изображений, свёрточная нейронная сеть должна быть предварительно обучена на различных моделях жилых домов (П-49, П-44, КОПЭ и т.д.). В качестве активационной функции свёрточная сеть использует сигмоидальную функцию.

$$f(x) = \frac{x}{|x| + \alpha}, \quad \alpha > 0. \quad (2)$$

Т.к. обучение нейронной сети может занимать до 1-2 лет, используются уже обученные нейронные сети, после чего они переобучаются на основе входных данных в виде жилых домов[5]. В статье будет использоваться сеть ConvNet. Для обучения различных

типов домов создаётся база данных жилых массивов, после чего она нейронная сеть обучается различать их между собой. На каждый тип домов требуется от 50 до 200 изображений[6].

После того обучения можно нейронная сеть готова к эксплуатации. Схема работы программы представлена на рисунке 2.

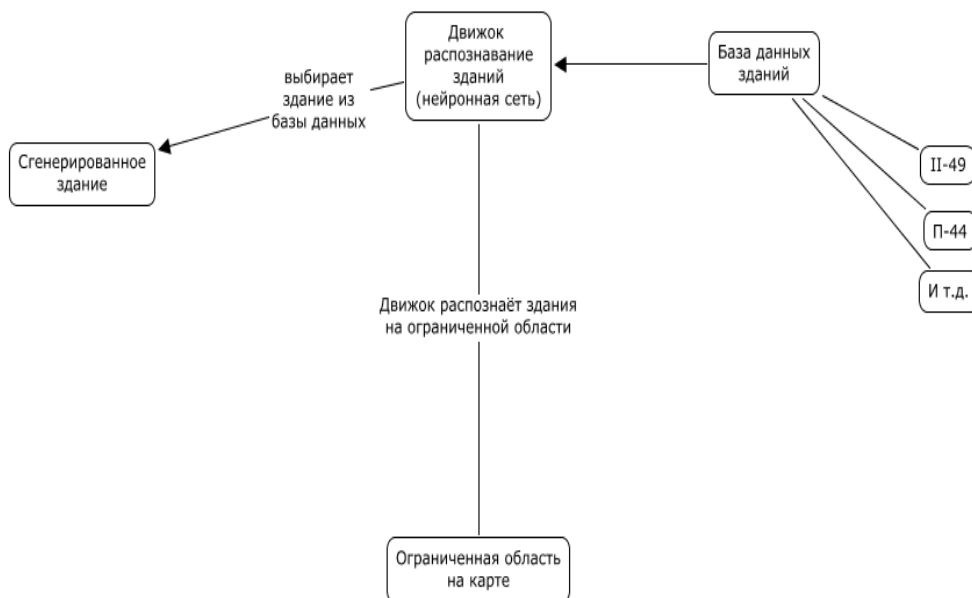


Рисунок 2 – Схема работы программы

Формат хранения объектов базы данных представлен в расширение .obj, а текстуры к ним в формате .jpg.

Изображения с ограниченной области на карте предоставляются в формате .bmp R5 G6 B5. На вход поступают 2 типа изображений: первый тип – фотография региона сверху, второй тип – изображение региона с точки зрения прохожего.

После загрузки региона, программа анализирует изображения на предмет имеющихся в базе данных зданий. В случае отсутствия здания, программа выдаст сообщение об отсутствие здания в базе данных. В качестве приемлемого результата, программа сгенерирует 3D представление здания на виде карты сверху. Результат распознавания приведён на рисунке 3.



Рисунок 3 – Результат распознавания здания

Программная реализация

Программная часть представлена на 2 языках программирования: C++ и Python. Модуль Python представляет собой саму нейронную сеть (файлы conv.py и utils.py), в то время как C++ отвечает за генерацию конечной сцены.

Файл conv.py содержит функции:

- conv_net(input_data, input_shape, n_classes, dropout_prob, reuse, is_training);
- create_model_function(learning_rate, dropout_rate, input_shape, num_classes = 2);
- train_and_evaluate(path, shape_x=128, shape_y=128, num_steps=2000, batch_size=200, learning_rate=0.001, dropout_rate=0.25, testing_percentage=30).

Функция conv_net задает топологию нейронной сети. Входные данные функции отвечают за:

- input_data – слой входных данных. tf.placeholders.;
- input_shape – истинная размерность входных данных (ширина x высота);
- n_classes – число предсказываемых классов;
- dropout_prob – вероятность, с которой каждая из связей последнего слоя будет выкинута при дропауте;
- reuse – надо ли повторно использовать уже созданные переменные (передается в вызов variable_scope);
- is_training – будет ли данная сеть использоваться для обучения (передается в слой дропаут для его активации).

На выход функция возвращает последний слой нейронной сети, который можно использовать для обучения и предсказания результатов.

Функция create_model_function задает Tensorflow модель (нейросеть + способ её обучения + способ ее оценивания), которую затем можно использовать в шаблоне TF Estimator:

- learning_rate – скорость обучения градиентного спуска;
- dropout_rate – частота выкидывания связей между слоями в дропауте;
- input_shape – истинная размерность входных данных (ширина x высота);
- num_classes – номер предсказываемого класса.

Функция train_and_evaluate является основной точкой входа в скрипт. Входные данные функции отвечают за:

- path – путь в файловой системе, где лежат картинки для обучения;
- shape_x – размер по горизонтали, к которому надо приводить все картинки;
- shape_y – размер по вертикали, к которому надо приводить все картинки;
- num_steps – число шагов обучения;
- batch_size – размер свёртки, на котором будет делаться каждый шаг обучения;
- learning_rate – скорость обучения градиентного спуска;
- dropout_rate – частота выкидывания связей между слоями в дропауте;
- testing_percentage – процент данных, используемых для обучения;

Файл utils.py предназначен для загрузки изображений, и содержит функции:

- load_images (image_dir, testing_percentage, shape);
- load_single_image (file_path);
- load_test_image(file_path).

Файл objloader.cpp отвечает за загрузку объектов в программу. Основные функции:

- Load – функция загрузки объекта расширения .obj;
- LoadTexture – функция загрузки текстуры объекта расширения .jpg/bmp;
- LoadMapData – функция загрузки фотографии региона сверху и изображения региона с точки зрения прохожего.

Файл `init.cpp` отвечает за выполнения Python-скрипта. Основные функции:

- `InitCNN` – функция запуска нейронной сети;
- `IfSuccessful` – функция проверки на успешность распознавания. Если нейросеть распознала здание, вызывается функция `GetRes`;

- `GetRes` – функция получения типа здания.

Файл `main.cpp` является точкой входа в программу.

Основные функции:

- `SDL_Init` – функция инициализации 3D представления сгенерированного здания;
- `SetVideoMode(1280, 960, 32, SDL_SWSURFACE | SDL_OPENGL)` – функция отвечающая за параметры окна отображения;

- `display()` – функция отображения 3D модели на экране;

- `init()` – функция исполнения нейронной сети.

Выводы

Анализ программного обеспечения с использованием нейронных сетей показал, что процесс обучения нейронной сети продолжителен. Однако, воспользовавшись имеющимися готовыми базами, можно эффективно применять нейронные сети для перевода двухмерного изображения в трёхмерное изображение. Исследования работ по созданию трёхмерной модели показал трудоёмкость этого процесса. Процесс значительно можно ускорить и удешевить, используя для таких целей сверточные нейронные сети.

Литература

1. Нейронные сети Google и дополненная реальность [Электронный ресурс]. – Режим доступа: <https://golos.io/ru--obrazovanie/@natasha/google-magic>
2. Google uses AI to sharpen low-res images [Электронный ресурс]. – Режим доступа: <https://www.engadget.com/2017/02/07/google-ai-image-enhancement/>
3. Барский, А.Б. Нейронные сети: Распознавание, управление, принятие решений / А.Б. Барский, А.В. Леоненков. – 2-е изд. – СПб.: БХВ, 2004. – 158 с.
4. Войтенко, А.В. Создание 3D модели города Красноярска, как часть картографического обеспечения градостроительного планирования и управления имущественным комплексом. АО «Кадастръёмка» / А.В. Войтенко, И.С. Кошечкин. // Иркутск, 2015. – С. 9-11.
5. Распознавание образов и новое поколение нейронных сетей Microsoft [Электронный ресурс]. – Режим доступа: <https://news.microsoft.com/ru-ru/microsoft-deep-residual-network>
6. Нейронную сеть научили отбирать потенциальные противораковые лекарства [Электронный ресурс]. – Режим доступа: <https://corp.mail.ru/ru/press/releases/9870>
7. DeepHD [Электронный ресурс]. – Режим доступа: <https://yandex.ru/company/technologies/deephd>
8. Нейронные сети: как они появились, зачем они нужны и чего от них ждать [Электронный ресурс]. – Режим доступа: <https://blogsibirix.ru/2018/01/30/neural>
9. Сегментация изображений [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/128768>
10. Сикорский, О.С. Обзор сверточных нейронных сетей для задачи классификации изображений / О.С. Сикорский. – М.: МГТУ им. Н.Э. Баумана, 2016. – 38 с.

УДК 004.021

СЕГМЕНТАЦИЯ ИЗОБРАЖЕНИЯ НА ОСНОВЕ МЕТОДА ОПТИМИЗАЦИИ РОЯ ЧАСТИЦ

Купоросов А.А.

Донецкий национальный технический университет

E-mail: kuporo090909@gmail.com

Аннотация:

Купоросов А.А. Сегментация изображения на основе метода оптимизация роя частиц. Сегментация цифровых изображений является одной из основных задач цифровой обработки изображений. Это процесс деления цифрового изображения на составляющие его объекты. В этой статье дается обзор методов сегментации изображений, основанных на методах кластеризации на основе Particle Swarm Optimization (PSO). PSO – это одна из новейших технологий сегментации цифровых изображений, вдохновленная природой.

Annotation:

Kuporosov A.A. Segmentation of the image based on the PSO method. Digital image segmentation is one of the main tasks of digital image processing. This is the process of dividing a digital image into its constituent objects. This article provides an overview of image segmentation methods based on Particle Swarm Optimization (PSO) based clustering methods. PSO is one of the newest digital image segmentation technologies, inspired by nature.

Введение

Цифровая обработка изображений – это обработка изображений, которые по своей природе являются цифровыми, с помощью цифрового компьютера. Обработка изображений мотивируется тремя основными приложениями: улучшение графической информации для восприятия человеком, обработка изображений для автономного приложения, эффективное хранение и передача. Процесс разделения цифрового изображения на несколько сегментов называется сегментацией цифрового изображения. Это одна из сложных и важных задач системы обработки изображений. Этот процесс выполняется для четкого представления изображения. Он часто используется для разделения изображения на отдельные области, которые идеально соответствуют различным объектам реального мира. Это важный шаг к анализу контента и пониманию изображений. Результатом процесса сегментации изображения является набор сегментов, которые объединяются, чтобы сформировать все изображение. Задачи сегментации изображений реального мира имеют множество целей, таких как минимизация функций, минимизация общего отклонения, минимизация частоты ошибок классификатора или максимизация связности и т. Д. Оптимизация роя частиц (PSO) – это эволюционная методика вычислений, разработанная Кенни и Эберхартом в 1995 году. PSO – это популяционный стохастический подход для решения задач непрерывной и дискретной оптимизации. При оптимизации роя частиц простые программные агенты, называемые частицами, перемещаются в пространстве поиска задачи оптимизации.

Цифровая обработка изображений – это обработка изображений, которые по своей природе являются цифровыми, с помощью цифрового компьютера. Обработка изображений мотивируется тремя основными приложениями:

- 1) улучшение графической информации для восприятия человеком;
- 2) обработка изображений для автономного приложения;
- 3) эффективное хранение и передача.

Процесс деления цифрового изображения на несколько сегментов называется сегментацией цифрового изображения. Это одна из сложных и важных задач системы

обработки изображений. Этот процесс выполняется для четкого представления изображения. Он часто используется для разделения изображения на отдельные области, которые идеально соответствуют различным объектам реального мира. Это важный шаг к анализу контента и пониманию изображений. Результатом процесса сегментации изображения является набор сегментов, которые объединяются, чтобы сформировать все изображение. Проблемы сегментации изображений реального мира имеют несколько целей, таких как минимизация функций, минимизация общего отклонения, минимизация частоты ошибок классификатора или максимизация связности и т. д.

Постановка задачи

Цели этого исследования:

- показать, что PSO может быть успешно использован для решения сложных задач в распознавании образов и обработке изображений;
- разработать инструмент, который может помочь исследователям в области классификации изображений без пристрастия проверить их алгоритмы, сравнить различные кластеры, алгоритмы и генерировать ориентиры;
- разработать эффективный алгоритм динамической кластеризации, который может найти «оптимальное» количество кластеров в наборе данных с минимальным вмешательством пользователя;
- разработать PSO-подход для решения проблемы квантования цветного изображения;
- разработать эффективный метод отбора конечных элементов на основе PSO для спектрального не смешивания данных многоспектральных изображений.

Методика основного PSO

Оптимизация роя частиц относится к классу методов разведки роя, которые используются для решения задач оптимизации. PSO имитирует поведение стая птиц. Значит, группа птиц случайным образом ищет еду в этом районе. В районе, где проводится обыск, есть только один кусок пищи. Все птицы не знают, где еда. Но они знают, как далеко еда в каждой итерации. Таким образом, лучший способ найти еду – следовать за птицей, которая находится ближе всего к еде. Стадное поведение – это поведение, которое проявляется, когда группа птиц, называемая стадом, добывает корм.

Каждая частица в PSO обновляется следующими двумя «лучшими» значениями:

pbest – каждая частица отслеживает свои координаты в пространстве решений, которые связаны с наилучшим решением (пригодностью), которое было достигнуто к настоящему моменту этой частицей. Это значение называется личным лучшим pbest.

gbest – это лучшее значение, полученное до сих пор любой частицей в окрестности этой частицы. Это значение называется Global Best, gbest.

Каждая частица пытается изменить свою позицию, используя:

- текущую позицию;
- текущую скорость;
- расстояние между текущей позицией и pbest;
- расстояние между текущей позицией и gbest.

После нахождения двух лучших значений частица обновляет свою скорость и положения с помощью следующих уравнений:

$$v[] = v[] + c1 * rand() * (pbest[] - present[]) + c2 * rand() * (gbest[] - present[]), \quad (1)$$

$$present[] = present[] + v[], \quad (2)$$

где $v[]$ – скорость частицы; $present[]$ – текущая частица (решение); $rand()$ – случайное число между (0; 1); $c1$ и $c2$ являются факторами обучения (обычно $c1=c2=2$).

Концепция модификации поисковой точки алгоритма PSO изображена на рисунке 1.

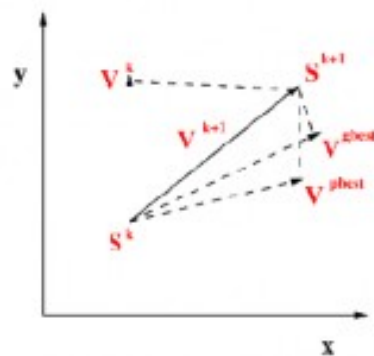


Рисунок 1 – Концепция модификации поисковой точки алгоритма PSO

Блок-схема алгоритма

На рисунке 2 приведена блок-схема алгоритма PSO.

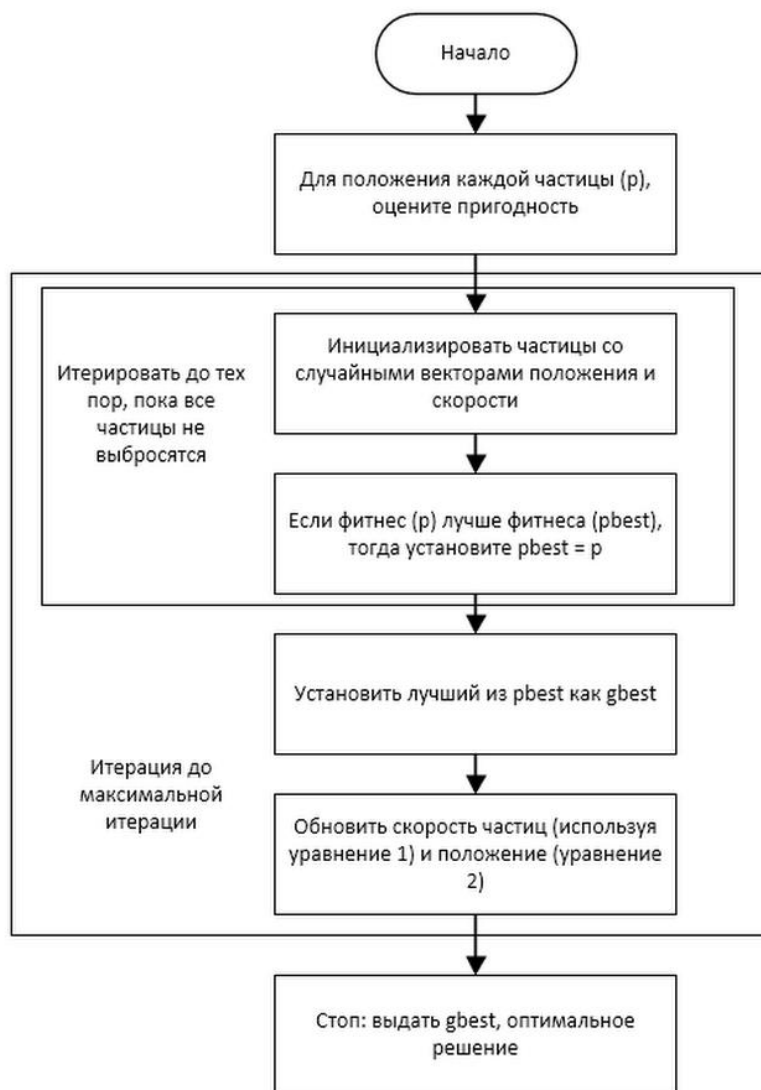


Рисунок 2 – Блок-схема алгоритма PSO

Применение этого алгоритма к изображению Проблема сегментации может быть упорядочена в следующем способе:

Шаг 1: Считать входное изображение для сегментирования.

Шаг 2: Выбрать метод PSO, который будет применен к этому изображению с определенным пороговым уровнем.

Шаг 3: Для каждой частицы в популяции обновить пригодность частицы в пространстве поиска и обновить лучшую частицу в пространстве поиска. Переместить частицу в популяции.

Шаг 4: Для каждой частицы сделать, если рой станет лучше, затем вознаградить рой, породив частицу: продлите жизнь роя / частицы.

Шаг 5: Для каждой частицы, если рой не улучшает свои характеристики, наказать рой: удалите рой / частицу или уменьшите срок годности роя.

Шаг 6: Расширение роя на нерест (рой считается для следующей итерации).

Шаг 7: Удалить «неуспешные» рои (рой никогда не попадет в область поиска) и сбросьте пороговый счетчик.

Результаты

PSO является естественной техникой вычислений и предоставляет ряд способов более эффективного и быстрого и точного решения реальных проблем. Сегментированные изображения на основе PSO, как правило, хорошо сегментированы на области однородного цвета и имеют перцептуальное значение для зрения человека и могут автоматически очень хорошо определять количество областей. Результат алгоритма, обсужденного в этой статье, показан на рисунках 3 и 4.



Рисунок 3 – Входное изображение

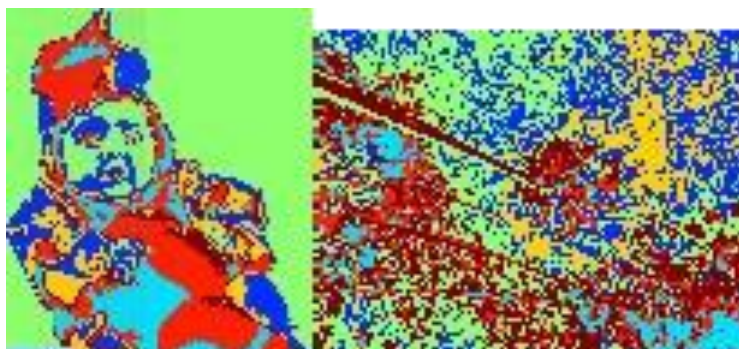


Рисунок 4 – Результат алгоритма на синтетических входных изображениях.

Выводы

Одной из наиболее сложных проблем обработки изображений является выделение контура объектов для их дальнейшего распознавания. Ключевой является задача выделения линий, проходящих на границах однородных областей.

Так как в большинстве случаев границы на изображениях выражены не столь контрастно, поэтому существующие на данный момент алгоритмы и операторы не обеспечивают эффективность получаемого результата. Часто результат обработки варьируется от разрывов контуров и линий (фрагментированность) до полного отсутствия границы. Одним из наиболее перспективных, на мой взгляд, является подход, основанный на методе оптимизации роем частиц.

В этой работе была описана техники разведки роя, а именно: рой частиц оптимизации. Были подробно описан алгоритм, основанный на этой парадигме. Новые алгоритмы улучшают базовый кластерный метод, который на самом деле основано на K-значении.

Литература

1. Терехин, А. В. Алгоритмы определения безразмерных признаков изображений проекций трехмерных объектов и их распознавание / Терехин А. В. // ВлГУ. – 2015. – 187 с.
2. Eberhart, R. C. A new optimizer using particle swarm theory [Текст] / R. C. Eberhart, J. Kennedy // Proceedings of the Sixth International Symposium on Micromachine and Human Science. – 1995. – pp. 39-43.
3. Eberhart, R. C. A discrete binary version of the particle swarm algorithm [Текст] / R. C. Eberhart, J. Kennedy // Proceedings of the IEEE International Conference on Systems. 1997. – pp. 4104-4108.
4. Wei, Z. Network Coverage Optimization Strategy in Wireless Sensor Networks Based on Particle Swarm Optimization [Текст] / Z. Wei, F. Zihao // Bachelor's Thesis in Electronics. – 2011. – P. 7.
5. Nyirarugira, C. Hand Gesture Recognition Using Particle Swarm Movement [Текст] / C. Nyirarugira, H. Choi, T. Kim // Mathematical Problems in Engineering. – 2016. – pp. 1-4.
6. Ladkany G. S. A Genetic Algorithm with Weighted Average Normally-Distributed Arithmetic Crossover and Twinkling [Текст] / G. S. Ladkany, T. B. Mohamed // Applied Mathematics. – 2012. – pp. 1220-1235.

УДК 004.89, 004.273

АРХИТЕКТУРА ИНТЕРНЕТА ВЕЩЕЙ НА БАЗЕ ТЕХНОЛОГИИ БЛОКЧЕЙНА

Матвеев М.О., Гудаев О.А.

Донецкий национальный технический университет, г. Донецк
кафедра искусственного интеллекта и системного анализа

E-mail: max_mat@inbox.ru

Аннотация:

Матвеев М.О., Гудаев О.А., Архитектура интернет вещей на базе технологии блокчейна. Рассмотрена модель формирования защищенных данных локальной подсистемы интернет вещей использующую технологию блокчейн. Определены цели и задачи достоверности данных поступающих из локальной подсистемы интернета вещей.

Annotation:

Matveev M.O., Gudaev O.A., Architecture of the Internet of things based on the blockchain technology. A model for the formation of protected data of the local subsystem of the Internet of Things using Blocking technology is considered. Defined goals and objectives of the reliability of data coming from the local subsystem of the Internet of Things

Общая постановка проблемы

Разработать архитектуру формирования безопасных данных генерируемых сенсорной подсистемой локальной части интернета вещей. Предложить компоненты и архитектуру использующую технологию блокчейна. Для гарантирования достоверности данных двух взаимных пиринговых подсистем интернет вещей.

Анализ существующих систем гарантирования безопасности данных.

Интернет вещей (IoT) представляет собой класс устройств, которые могут обмениваться любым видом данных между собой, тем самым создавая сеть взаимодействия. Использование блокчейна может гарантировать сохранность и целостность данных в IoT, а также обеспечивать надежную систему безопасности [4]. Компания Chronicled в августе прошлого года запустила блокчейн-платформу для Интернета вещей, которая позволяет компаниям и производителям регистрировать и подтверждать физические предметы в блокчейн-сети. А компания Filament предлагает ряд собственных программных и аппаратных решений для управления промышленными системами и оборудованием. В основе разработок компании лежат принципы децентрализации, криптографической защиты и автономности.

Основой блокчейна является децентрализованное доверие - неизменный источник правды. Сегодня данные IoT нельзя полностью доверять за пределами домена владельца данных из-за невозможности проверить, что данные не были обработаны перед их отправкой, продажей или использованием другими сторонами [3]. Например, автономные автомобильные стартапы и гиганты совместного использования поездок, такие как Uber и Lyft, не имеют решения для совместного использования надежных картографических данных или данных поездок; вместо этого они собирают и хранят аналогичные наборы данных независимо друг от друга.

Доверенные среды выполнения (TEE) в современных устройствах IoT могут свидетельствовать о том, что данные поступили с определенного устройства, и как только эти данные сохраняются в блокчейне, они являются неизменяемыми и отслеживаемыми. Комбинация безопасных устройств IoT и блокчейна представляет уникально чистый

источник данных IoT, который, как известно, поступает из определенного источника и не подвергается манипуляциям. Скоро появятся новые бизнес-модели, основанные на этих надежных данных IoT - данные можно покупать, продавать, делиться и использовать среди ненадежных сторон.

Согласно ежеквартальным отчетам Kaspersky Lab, увеличилась длительность и сложность DDoS-атак, с которыми компании столкнулись в 2017 году. Так, во втором квартале 2017 года состоялась самая продолжительная атака, которая длилась 227 часов. В частности, эксперты отметили рост количества атак, направленных на площадки, проводящих ICO.

Блокчейн-стартап Gladius разрабатывает проект по запуску децентрализованной одноранговой сети, работающей без главного сервера, которая обеспечит защиту от DDoS атак. Gladius позволяет пользователям монетизировать неиспользованную пропускную способность своих интернет-сетей, объединяя их в децентрализованную CDN (Content Delivery Network). Благодаря высокой пропускной способности сеть отразит DDoS-атаки, а пользователи смогут подключаться к безопасным пулам защиты[2].

Компоненты блокчейна для взаимодействия с интернет вещей.

Общая схема системы будет предоставлена на рисунке 1.

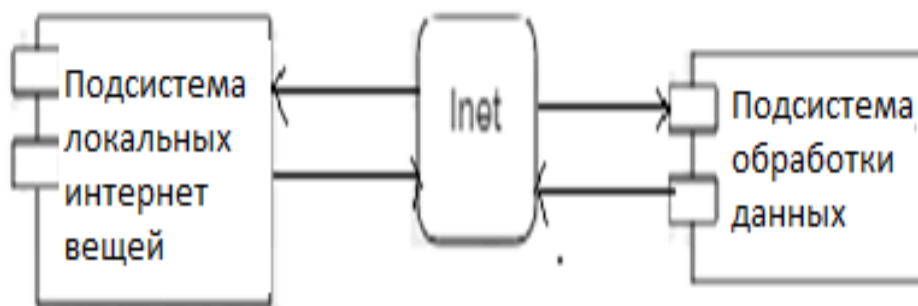


Рисунок 1. Общая схема компонентов системы.

Компонентами локальных интернет вещей являются две подсистемы(см рис.1).

Первая из них называется - подсистема локальных интернет вещей, в её состав входят:

- Сервер управления и сбора телеметрических данных;
- Модуль генерации блокчейн- подписи данных на основе онтологического словаря связанных данных bliked DATA;
- Модуль сопоставленных локальных и глобальных знаний связанных данных;

Второй подсистемой является подсистема обработки данных, в её состав входят:

- Информационные хранилища;
- Модуль блокчейн кодирования онтологических знаний ключевых слов;
- Модуль сопоставления и проверки на достоверность данных локальной подсети;

Из общей схемы наблюдается, что связующим звеном между подсистемой обработки данных и подсистемы локальных интернет вещей является интернет[1]. Подробно о работах подсистем, пронумерованных на схеме, будут рассмотрены в следующем подразделе.

Архитектура локальных интернет вещей

Из диаграммы компонентов (см.рис 2) можно увидеть архитектуру взаимодействие компонентов локальных интернет вещей

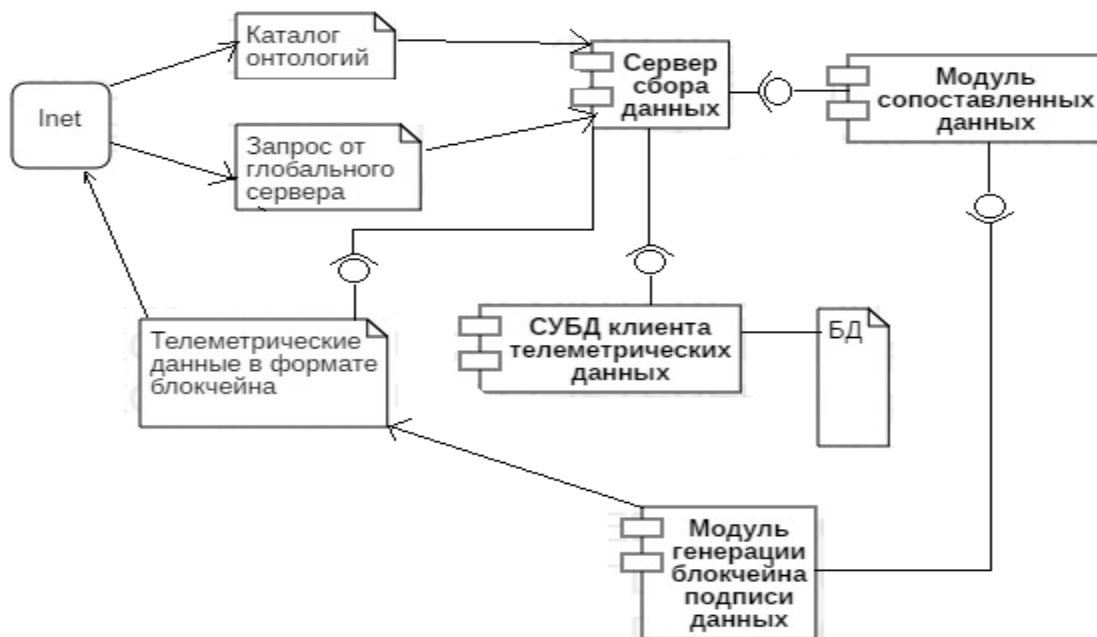


Рисунок 2. Диаграмма компонентов локальных интернет вещей.

Как видно из диаграммы компонентов (см.рис 2) после обработки запроса, сервер обращается к онтологическому словарю, модуль генерирует подписи данных связанные с базой данных. После этого он сопоставляет блокчейн подписи с телеметрическими данными и отправляет ответ обратно на сервер. После чего данные попадают в сеть. Таким образом происходит безопаснейшее шифрование данных интернет вещей.

Архитектура подсистемы обработки данных

Из диаграммы компонентов (см.рис 3) можно увидеть архитектуру взаимодействие компонентов подсистемы обработки данных.



Рисунок 3. Диаграмма компонентов дсистемы обработки данных.

Как видно из диаграммы компонентов (см.рис 3) после обработки запроса, сервер

обрабатывает данные от подсистемы 1. Далее он сопоставляет их в модуле сопоставления данных и блокчейн кодирования и отправляет в СУБД хранения данных. После запроса данные можно получить из облачного хранилища. Таким образом происходит безопаснейшая обработка данных.

Наиболее непосредственные случаи использования блокчейна и IoT для предприятия будут сосредоточены на многостороннем доверии и обмене данными, что позволяет использовать различные варианты снижения затрат и соответствия нормативным требованиям. Например, использование надежных данных для автоматизации многосторонних рабочих процессов и подтверждения соответствия определенным правилам и нормам [5]. Прежде чем предприятия будут использовать и принимать публичные блокчейны / DApps с токеновой экономикой, компании блокчейнов должны научиться разрабатывать решения, которые дополняют существующие системы / процессы, а не заменяют их (пока).

Выводы

Исследовал уровни архитектуры клиента локальных устройств интернет вещей кодирующего телеметрические данные технологией блокчейна для гарантирования их достоверности. Рассмотрел архитектуру глобального сервера обработки данных от подсистемы клиента.

Литература

1. Омельченко, Д. А. Технология блокчейн прошлое, настоящее, будущее [Текст] / Д. А. Омельченко, К. А. Ковалева, // Сб. ст. межд. конф. «Проблемы и перспективы в международном трансфере инновационных технологий». – Уфа: ООО «АМИ», 2018. - С. 149-152.
2. Купцева, В. В. Информационные системы в экономике с применением блокчейна [Текст] / В. В. Купцева // Сб. ст. Аллея науки. – Поволжск: Издательский центр «Quantum», 2018. - С. 882-865.
3. Болотова, Л. С. Системы искусственного интеллекта: модели и технологии, основанные на знаниях [Текст] / Л. С. Болотова // учебник «Финансы и статистика». – Москва: ООО «Литсвет», 2015. - С. 411-419.
4. Матвеев, М. О. Архитектура искусственного интеллекта на базе технологии блокчейна / М. О. Матвеев, О. А. Гудаев // Материалы IX Международной научно-технической конференции «Информатика, управляющие системы, математическое и компьютерное моделирование» (ИУСМКМ-2018), 22-24 мая 2018. – Донецк: ДонНТУ, 2018. – С. 29-32.
5. Курейчик, В. М. Параллельный генетический алгоритм. Модели и проблемы построения [Текст] / В. М. Курейчик, Д. С. Кныш // Сб. науч. ст. V Междунар. науч.-практ. Конф. «Интегрированные модели и мягкие вычисления в искусственном интеллекте» - Москва: «Физматлит», 2016. - С. 41 - 51.

УДК 004.424

АРХИТЕКТУРА ПРОГРАММНОГО КОМПЛЕКСА РЕЗЕРВНОГО КОПИРОВАНИЯ ДАННЫХ

А.И. Ольшевский, В.С. Нестеренко

Донецкий национальный технический университет, г. Донецк
кафедра искусственного интеллекта и системного анализа

E-mail: a_olshevskiy@mail.ru, nes.vadim.s@gmail.com

Аннотация:

Ольшевский А.И., Нестеренко В.С. Архитектура программного комплекса резервного копирования данных. В данной статье рассмотрены изменения в стандартной архитектуре. Определена структура программного комплекса, разделены задачи и функции клиентской и серверной части. Предложен алгоритм анализа статистических данных для выявления временных промежутков наименьшей загрузки ЭВМ и алгоритм доверия к хранилищам.

Ключевые слова: *архитектурная модель* программного комплекса, корпоративные сети, резервное копирование данные, загрузки ЭВМ, статистические данные.

Annotation:

Olshevsky A.I., Nesterenko V.S. Architecture software backup data. This article describes the changes in the standard architecture. The structure of the software package was determined, the tasks and functions of the client and server parts were separated. An algorithm for analyzing statistical data to identify the time intervals of the smallest workload of a computer and an algorithm for trusting storages is proposed.

Keywords: *architectural model of the software package, corporate networks, backup data, computer workload, statistical data.*

Постановка проблемы

Актуальными можно выделить две проблемы: отсутствие приемлемых систем резервного копирования данных с трёхуровневой архитектурой для корпоративных сетей (на сегодняшний день существует единичные платные решения с множеством условностей) и разработка алгоритма гибкого расписания резервного копирования на основе статистических данных «загруженности» работы ЭВМ в течении некоторого периода времени (нагрузки центрального процессора, сетевой активности и текущей работы жёсткого диска).

Внедрение адаптивного статистического алгоритма по управлению расписанием в программное обеспечение является уникальным решением на сегодняшний день. Еженедельное, ежемесячное и ежеквартальное резервное копирование подразумевает создание полной копии всех данных. Обычно оно выполняется тогда, когда копирование большого объёма данных не влияет на работу организации. Данное решение позволит не только оптимизировать загрузку ЭВМ пользователя, но и помочь в работе специалистам, отвечающим за сохранность данных.

Проанализировав современные актуальные разработки в сфере хранения и передачи данных, а так же рассмотрев проблемы и программные решения резервного копирования данных, было выделено направление по разработке новой архитектуры программного обеспечения.

Цель работы – повышение надежности и отказоустойчивости, увеличения быстродействия системы и уменьшая нагрузку на пользовательские ЭВМ, за счет статистического анализа и составления расписания резервного копирования.

Трёхуровневая архитектура

Исследования трёхуровневой архитектуры программного обеспечения и апробация разработанного комплекса, с учётом расширения дополнительного функционала и некоторых изменений актуальных для пользователя, подверглись некоторым изменениям [1-3].

В программе используется модифицированная трёхуровневая архитектура, которая включает в себя такие компоненты (слои): клиент; сервер, подключенный к клиентским приложениям и хранилище с которым работает этот сервер. Изменения архитектуры представляет собой возможность работы программного комплекса в локальной сети, что является актуальным вопросом, в связи с крайне малым количеством других программ с данным функционалом. Стоит отметить, что для каждого слоя представлены свои изменения и доработки. На рисунке 1 изображена трёхуровневая архитектура программного комплекса [4].



Рисунок 1 – Трёхуровневая архитектура программного комплекса

Первым слоем в данной архитектуре является клиент, который в данном случае представляет собой приложение на ЭВМ пользователя по обработке данные, передаваемые с сервера, посредством контроля через веб-интерфейс [5]. ЭВМ пользователя, регистрируется к его аккаунту на стороне сервера. Принимает задачи, и в соответствии с ними по расписанию отправляет копию данных на указанные пользователем хранилища с заданными пользователем настройками, а также сообщает серверу о состоянии каждой задачи при их выполнении. Модификация данного слоя представляет собой хранение сообщений, посылаемых серверу приложений при отсутствии подключения к сети, и отправки сохранённых сообщений при подключении к сети.

Второй слой представляет собой сервер с веб-интерфейсом, через который идёт подвязка клиентов к аккаунту пользователя с помощью уникального ключа, настройка задач резервного копирования, управление ими, а также просмотр статуса и истории задач резервного копирования. Задача имеет такие настройки как: выбор файлов, архивирование,

выбор места хранения, расписание, уведомление о состоянии, шифрование, и история задач [6].

Модификация данного слоя представляет собой возможность управления хранимыми данными, предоставляя пользователю выбор места хранения (на сервере-хранилище в локальной сети или в хранилище сети Интернет), а также возможность распределенного хранения и считывания из разных хранилищ данных.

Сервер баз данных (слой данных) обеспечивает хранение данных и выносятся на отдельный третий уровень. В программном средстве он реализуется средством системы управления базами данных MongoDB, подключение к этому компоненту обеспечивается только с уровня сервера приложений [7]. Модификация данного слоя представляет собой распределенное хранение данных для обеспечения большей надежности и скорости считывания данных.

Программный комплекс является открытым, что позволяет делать его код общедоступным для просмотра и изменения в разработке данного программного обеспечения [8].

Алгоритм анализа статистических данных для выявления временных промежутков наименьшей загруженности ЭВМ

Составляя задачу резервного копирования данных, специалисту необходимо указывать время начала выполнения данной задачи, определяя время наименьшей загруженности ЭВМ для максимизации эффективности работы ЭВМ внутри сети, выполняющие различные задачи. Специалисты, как правило, ставят время выполнения задач по резервному копированию или техническому обслуживанию на ночь, т.е. во временное окно наименьшей загруженности, чтобы никоим образом не влиять на работу иных программ и задач, не понижая их эффективность работы. Однако в крупных компаниях, где постоянно идёт обработка большого количества различных данных, такие временные окна специалистам определять трудно, в связи, с чем приходится обращаться к сторонним средствам анализа работы инфраструктуры предприятия, для определения времени наименьшей загруженности ЭВМ и серверов предприятия [9-10].

Данный алгоритм позволяет определить нагрузку центрального процессора, жёсткого диска и нагрузку канала сети-Интернет для каждой ЭВМ. По каждому, выставленному пользователем, временному промежутку текущее состояние сохраняется на протяжении дня. Пользователь может посмотреть статистику в виде статистического графика по каждому дню или усредненную статистику за определённый период времени (неделя, месяц и т.д.).

В планах дальнейшего улучшения алгоритма – его использование на стороне сервера для автоматической сортировки порядка задач по времени выполнения во временных окнах, в зависимости от размера файлов. Определяя загруженность канала локальной сети или сети-Интернет алгоритм позволит выполнять резервные копии последовательно, сохраняя данные с разных ЭВМ последовательно, а не одновременно, что снизит риск повреждения или потери данных. На рисунке 2 представлен усреднённый график статистических данных загруженности ЭВМ пользователя за неделю, с градуировкой в один час, с явно видимым временным промежутком наименьшей загруженности, в который следует выполнять резервное копирование данных.

Стоит отметить также и некоторые проблемы алгоритма, для решения которых требуется дополнительные исследования для полноты реализации в программном продукте.

Первая проблема – определение времени выполнения резервного копирования, с учётом загруженности ЦП, жёсткого диска и канала сети-Интернет. Может возникнуть ситуация, когда выделенного временного промежутка не хватит на выполнение задачи резервного копирования, в этом случае придётся «жертвовать» ресурсами ЭВМ пользователя вне временного окна с наименьшей загруженностью. Однако, если определить примерное время завершения выполнения резервного копирования данных с учётом нагрузки

вышеперечисленных компонентов, то некоторые задачи можно перенести на более нагруженные временные промежутки, но не на достигающие пика нагрузки. Также задачи можно распределить на несколько дней, но тут встаёт вопрос приоритета задач, реализация которого является следующей проблемой.



Рисунок 2 – Усреднённый график статистических данных загруженности ЭВМ пользователя

Вторая проблема – приоритет задач. Резервное копирование большого количества данных может занимать много времени, также, как и резервное копирования множества небольших данных. Здесь перед пользователем стоит вопрос, каким файлам назначить больший приоритет, т.е. выполнить копирование в первую очередь, а каких в последнюю, возможно выйдя за временной промежуток наименьшей загруженности ЭВМ.

Третья проблема – загруженность сетевого хранилища. Если множество пользовательский ЭВМ будут в одно время выполнять резервное копирование данных на сетевое хранилище, то помимо случая полной загруженности пропускной способности сетевого канала, скорость записи на накопитель в сетевом хранилище ограничена, следовательно помимо варианта записи данных на разные накопители, другим вариантом решения проблемы является реализация алгоритма на стороне сервера. Сервер, получая статистические данные с ЭВМ пользователей, должен определять как временные промежутки наименьшей загруженности ЭВМ, так и сетевого хранилища.

Надежность и достоверность хранилища данных

При выполнении резервного копирования нередко происходят случаи потери сетевого соединения в виду различных факторов. При разработке программного комплекса была поставлена задача приоритета автоматизации, в том числе и для решения подобного вида ситуаций. Был разработан алгоритм доверия к хранилищам: допустим, в программе подключено два сетевых хранилища, выделен параметр «доверие», определяющий количество резервных копий в процентном соотношении, хранимое на хранилище. Предположим, что доверие к обоим хранилищам равно десяти, но к первому хранилищу иногда теряется сетевое соединение. При выполнении резервного копирования на оба сетевых хранилища записываются данные, но при разрыве соединения доверие снижается. Поэтому, выполняется повтор отправки данных, и процентное соотношение хранимых данных на хранилищах будет следующим – на первом будет храниться 40% всех данных, когда на втором 60%. По прошествии какого-то количества времени доверие

восстанавливается, если сетевое соединение при выполнении резервного копирования не теряется, или уменьшается, если соединение теряется.

В данном алгоритме можно выделить следующее потенциальное развитие – проверка целостности файлов и перенос всех данных на другое хранилище, если первое находится в критическом состоянии.

Выводы

Определена архитектура программного комплекса, разделены задачи и функции клиентской и серверной части. Выделены основные изменения стандартной архитектуры. Предложен алгоритм анализа статистических данных для выявления временных промежутков наименьшей загруженности ЭВМ, с целью генерации расписания резервного копирования и за счет этого повышение эффективности работы системы. Разработан алгоритм определения доверия к хранилищам для повышения надежности и достоверности данных. Определены основные недостатки данных алгоритмов и их потенциальное развитие [11].

Литература

1. Ольшевский А.И. Система удаленного резервного копирования данных для корпоративных сетей [Текст] / А.И. Ольшевский, В.С. Нестеренко // Сб. мат. «Программная инженерия: методы и технологии разработки информационно-вычислительных систем» (ПНИВС – 2018). – Донецк : ДонНТУ, 2018. – С. 88-92
2. Ольшевский А.И. Исследование и проектирование программного комплекса удаленного резервного копирования данных [Текст] / А.И. Ольшевский, В.С. Нестеренко // Сб. мат. «Информатика, управляющие системы, математическое и компьютерное моделирование» (ИУСМКМ – 2018). – Донецк : ДонНТУ, 2018. – С. 61-64
3. Бычкова Е.В. Программное средство создания резервных копий данных [Текст] / Е.В. Бычкова, В.С. Нестеренко // Сб. науч. тр. «Информатика, управляющие системы, математическое и компьютерное моделирование» в рамках III форума «Инновационные перспективы Донбасса» (ИУСМКМ – 2017). – Донецк : ДонНТУ, 2017. – С. 381-384
4. Трёхуровневая архитектура [Electronic resource] / Интернет-ресурс. – Режим доступа : [www/ URL: https://ru.wikipedia.org/wiki/Трёхуровневая_архитектура](http://www/URL:https://ru.wikipedia.org/wiki/Трёхуровневая_архитектура). – Загл. с экрана.
5. Веб-интерфейс [Electronic resource] / Интернет-ресурс. – Режим доступа : [www/ URL: https://ru.wikipedia.org/wiki/Веб-интерфейс](http://www/URL:https://ru.wikipedia.org/wiki/Веб-интерфейс). – Загл. с экрана.
6. Buschmann F. Pattern-Oriented Software Architecture Volume 1: A System of Patterns / Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal. – Chichester : John Wiley & Sons, 1996. – 476 p.
7. MongoDB [Electronic resource] / Интернет-ресурс. – Режим доступа : [www/ URL: https://www.mongodb.com](http://www/URL:https://www.mongodb.com). – Загл. с экрана.
8. Xie Z. Open Source Software Foundation / Zhensheng Xie. – Saarbrucken : LAP Lambert Academic Publishing, 2010. – 128 p.
9. Dasgupta S. Algorithms / Sanjoy Dasgupta, Christos H. Papadimitriou, Umesh Vazirani. – [1st Edition]. – New York : McGraw-Hill Education, 2006. 336 p.
10. Sedgewick R. Algorithms / Robert Sedgewick, Kevin Wayne. – [4th Edition]. – Boston : Addison-Wesley Professional, 2011. – 976 p.
11. Steven S. The Algorithm Design Manual / Steven S. Skiena. – Berlin : Springer, 2008. – 748 p.

УДК 004.55

ИССЛЕДОВАНИЯ РАЗРАБОТКИ ДИНАМИЧЕСКОГО ЯЗЫКА ЖЕСТОВ ЧЕЛОВЕКА ДЛЯ УПРАВЛЕНИЯ МУЛЬТИМЕДИЙНЫМ КОНТЕНТОМ

А.Д. Гетманова, О.М. Копытова

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: anyutkagetmanova@gmail.com, omkop@list.ru

Аннотация

Гетманова А.Д., Копытова О.М. Исследования разработки динамического языка жестов человека для управления мультимедийным контентом. Рассмотрены этапы распознавания изображений и основные операции: предварительная обработка, выделение признаков объекта и фильтрация.

Abstract

Getmanova A., Kopytova O. Research on the development of a dynamic human sign language for managing multimedia content. The stages of image recognition and basic operations are considered: preliminary processing, selection of object features and filtering.

Введение

В настоящее время компьютерное зрение применяется во многих областях. Данная технология используется не только в компьютерных играх, аэросъемке, создании уличных карт, но и при проектировании систем управления беспилотными аппаратами, обработки биомедицинских анализов.

Компьютерное зрение – теория и технология создания машин, которые могут производить обнаружение, отслеживание и классификацию объектов на получаемом из сенсорных источников видеосигнале [1].

Технология компьютерного зрения используется в различных системах, связанных со считыванием и интерпретацией визуальной информации в обрабатываемые компьютером команды и данные. Практическая разработка системы компьютерного зрения предполагает реализацию трех основных компонентов процедуры распознавания образов:

- предварительная обработка изображений;
- выделение признаков объекта;
- фильтрация изображений.

Цель исследования распознавания жестов заключается в создании системы, которая может определить конкретные человеческие жесты и использовать их для передачи информации или для управления устройством.

Задача данной работы состоит в рассмотрении этапов системы распознавания и основных компонентов процедуры.

Актуальность: Разработка эффективных алгоритмов распознавания динамического языка жестов актуальна для дальнейшего использования в различных мультимедийных системах. Также данные алгоритмы могут использоваться в системах дополненной виртуальной реальности.

Типы задач в распознавании

Системы распознавания имеют типичную функциональную схему, изображенную на рисунке 1. Входные данные, подлежащие распознаванию, подаются на вход системы и подвергаются предобработке с целью их преобразования в необходимый для следующего этапа вид или для выделения из них необходимых характерных признаков. Далее на этапе

принятия решения над обработанным массивом данных производится ряд вычислений и на основе их результатов формируется ответ, содержащий ожидаемые от системы сведения о входных данных. Содержание входных и выходных данных определяется назначением системы.

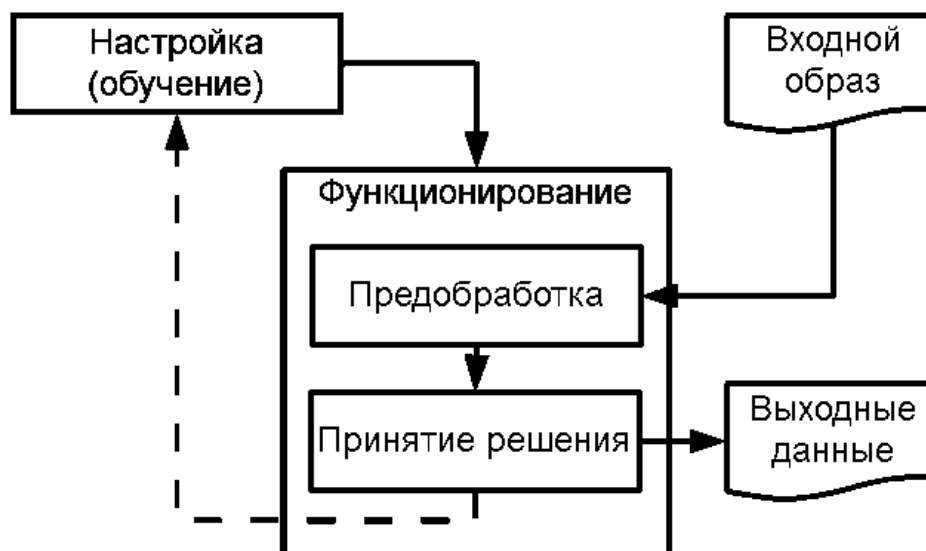


Рисунок 1 – Типичная функциональная схема системы распознавания

Кроме описанных этапов функционирования системы распознавания предусматривают свою настройку на множество возможных входных данных; этот этап называют этапом обучения системы. Целью обучения системы является формирование в её памяти набора сведений, необходимых для распознавания предполагаемого класса входных данных.

На этапе предобработки решается задача создания формализованного описания объектов распознавания, пригодного для использования алгоритмами собственно распознавания. Как правило, исходные данные о наблюдаемых объектах представлены в форме, непригодной непосредственно для распознавания. Это могут быть растровые изображения, звуковые файлы, статистические данные (числовые наборы), видеозаписи. Некоторые алгоритмы распознавания требуют более высокоуровневого представления. Это приводит к необходимости произвести одно или более преобразований исходных данных, переходя от кода 0 к коду 1, 2 и т.д. В качестве примера можно рассмотреть процедуру сегментации изображения, т.е. выделения на нём одноцветных областей.

Этап принятия решения является наиболее значимым в цикле работы системы распознавания с точки зрения её характеристики в целом. Т.е. задача, решаемая на данном этапе, во многом определяет назначение системы. Кроме того, для обеспечения возможности системы выполнять качественное принятие решений выдвигается ряд требований к этапу обучения. Наконец, как уже отмечалось выше, алгоритмы этапа принятия решений требуют необходимой предобработки входных данных.

Предварительная обработка изображений

Операции распознавания на изображениях определенных объектов, как правило, предваряются обработкой изображений для создания условий, повышающих эффективность и качество выделения и распознавания искомым или изучаемых объектов. Методы предварительной обработки зависят от задач исследований, довольно разнообразны и могут включать, например, выделение наиболее информативных фрагментов, их увеличение, получение 3-мерных изображений, цветокартирование, реализация высокого

пространственного разрешения, повышение контрастного разрешения, улучшение качества изображений и т.п.

Выделение признаков объекта

Выделение признаков позволяет упростить реализацию распознавания или идентификации объектов. При выборе наиболее информативных признаков необходимо учитывать как свойства самих объектов, так и возможности разрешающей способности первичных формирователей сигнала изображения. Выделение признаков проведем на примере обработки монохромных (однослойных) изображений. В цветных изображениях рассмотренные алгоритмы можно применять к каждому цвету в отдельности.

При обработке предпочтительными являются следующие признаки объектов:

- площадь и периметр изображения объекта;
- размеры вписанных простейших геометрических фигур (окружностей, прямоугольников, треугольников и др.);
- число и взаимное расположение углов;
- моменты инерции изображений объектов.

Важной особенностью большинства геометрических признаков является инвариантность относительно разворота изображения объекта, а путем нормирования геометрических признаков друг относительно друга достигается инвариантность относительно масштаба изображения объекта.

Фильтрация изображений

Обычно изображения, сформированные различными информационными системами, искажаются действием помех. Это затрудняет как их визуальный анализ человеком-оператором, так и автоматическую обработку в ЭВМ. При решении некоторых задач обработки изображений в роли помех могут выступать и те или иные компоненты самого изображения. Например, при анализе космического снимка земной поверхности может стоять задача определения границ между ее отдельными участками – лесом и полем, водой и сушей и т.п. С точки зрения этой задачи отдельные детали изображения внутри разделяемых областей являются помехой.

Ослабление действия помех достигается фильтрацией. При фильтрации яркость (сигнал) каждой точки исходного изображения, искаженного помехой, заменяется некоторым другим значением яркости, которое признается в наименьшей степени искаженным помехой. Изображение часто представляет собой двумерную функцию пространственных координат, которая изменяется по этим координатам медленнее (иногда значительно медленнее), чем помеха, также являющаяся двумерной функцией. Это позволяет при оценке полезного сигнала в каждой точке кадра принять во внимание некоторое множество соседних точек, воспользовавшись определенной похожестью сигнала в этих точках. В других случаях, наоборот, признаком полезного сигнала являются резкие перепады яркости.

Однако, как правило, частота этих перепадов относительно невелика, так что на значительных промежутках между ними сигнал либо постоянен, либо изменяется медленно. И в этом случае свойства сигнала проявляются при наблюдении его не только в локальной точке, но и при анализе ее окрестности. Заметим, что понятие окрестности является достаточно условным. Она может быть образована лишь ближайшими по кадру соседями, но могут быть окрестности, содержащие достаточно много и достаточно сильно удаленных точек кадра. В этом последнем случае, конечно, степень влияния далеких и близких точек на решения, принимаемые фильтром в данной точке кадра, будет совершенно различной.

Таким образом, идеология фильтрации основывается на рациональном использовании данных как из рабочей точки, так и из ее окрестности.

Задача заключается в том, чтобы найти такую рациональную вычислительную процедуру, которая позволяла бы достигать наилучших результатов. Общепринято при решении этой задачи опираться на использование вероятностных моделей изображения и

помехи, а также на применение статистических критериев оптимальности. Причины этого понятны – это случайный характер, как информационного сигнала, так и помехи и это стремление получить минимальное в среднем отличие результата обработки от идеального сигнала. Многообразие методов и алгоритмов связано с большим разнообразием сюжетов, которые приходится описывать различными математическими моделями. Кроме того, применяются различные критерии оптимальности, что также ведет к разнообразию методов фильтрации. Наконец, даже при совпадении моделей и критериев очень часто из-за математических трудностей не удается найти оптимальную процедуру. Сложность нахождения точных решений порождает различные варианты приближенных методов и процедур.

Выводы

Таким образом, в ходе исследования были рассмотрены этапы системы распознавания и основные компоненты процедуры. Стоит отметить, что разработанная система распознавания динамического языка жестов имеет большое разнообразие вариантов использования в различных сферах деятельности и областях.

Литература

1. Журавель И.М. Краткий курс теории обработки изображений / И.М. Журавель – М. : АН СССР, 1987. – 392 с.
2. Можейко В.И. Автоматическое сопровождение объектов в компьютерных системах обработки изображений / В.И. Можейко, В.Т. Фисенко, Т.Ю. Фисенко // Оптический журнал. – 2007 – №11. – С. 39-46.
3. Прэтт У. Цифровая обработка изображений / У. Прэтт – М. : Мир, 1982. – 791 с.
4. Марр Д. Зрение. Информационный подход к изучению представления и обработки зрительных образов / Д. Марр – М.: Радио и связь, 1987. – 637 с.
5. Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс.; [пер. с англ.]. – М.: Техносфера, 2005. – 1072 с.
6. Яне Б. Цифровая обработка изображений / Б. Яне – М.: Техносфера, 2007. – 564 с.

УДК 004.021

МОДЕЛИРОВАНИЕ И АНАЛИЗ МИМИЧЕСКИХ ПРОЯВЛЕНИЙ ЭМОЦИЙ НА ЛИЦЕ ЧЕЛОВЕКА

Пойда А.А.

Донецкий национальный технический университет

E-mail: artembratok@gmail.com

Аннотация:

Пойда А.А. Моделирование и анализ мимических проявлений эмоций на лице человека. Работа посвящена разработке методов, моделей, алгоритмов и компьютерных средств для моделирования и распознаванию эмоционального состояния лица человека.

Annotation:

Poyda A.A. Modeling and analysis of facial expressions of emotions on a person's face. The work is devoted to the development of methods, models, algorithms and computer tools for modeling and recognizing the emotional state of a person's face.

Введение

Методы и алгоритмы анализа и синтеза эмоционального состояния лица человека являются составной частью систем и средств искусственного интеллекта, направленных на исследование, создание и внедрение алгоритмических и программно-аппаратных систем и комплексов с элементами искусственного интеллекта на основе моделирования интеллектуальной деятельности человека. Моделирование и распознавание эмоций лица является актуальным и важным направлением исследований с целью создания систем компьютерного распознавания и синтеза зрительных образов. Невербальная мимическая передача информации человеком стала предметом интенсивных исследований.

Использование контурных моделей на базе NURBS-кривых

Для анализа мимических проявлений эмоций, имея в виду формальную модель эмоций состояния человека, предлагается оригинальная технология. Основная ее идея заключается в том, чтобы предложить алгоритм автоматического получения гибких шаблонов (на базе NURBS-кривых) контуров носа, бровей, глаз и рта. Требуется разработать:

- метод, который даёт возможность выделить наиболее важные части лица человека, а именно: нос, брови, глаза и рот;
- алгоритм получения точечных кривых, соответствующих контурам лица, которые будут анализироваться;
- алгоритм аппроксимации полученных точечных кривых с помощью NURBS-кривых;
- алгоритм анализа мимических проявлений эмоций, которые используются в качестве векторного пространства.

Метод получения контуров основных частей лица

Данный метод дает возможность выделить по сетке высот наиболее важные части лица, а именно: нос, брови, глаза, губы и щеки.

Определение опорных точек рта. Для отображения открытости рта необходимо точно определить его положение. Крайние точки можно найти программными методами, но существуют модели голов, которые распознать с необходимой точностью достаточно трудно. Это сделать можно вручную, что не требует много времени и не будет давать значительных ошибок. Обозначим их соответственно (X_n, Y_n) – крайняя правая и (X_l, Y_l) – крайняя левая точка. С помощью этих точек будем определять все части лица.

Построение частей профиля, которые проходят через нос. При построении профиля носа используем несколько особенностей, которые отличают его среди всех частей лица:

- 4) из всех точек лица высшая точка по z находится на носу;
- 5) при движении вдоль профиля ото рта до носа мы наталкиваемся на стремительный перепад высот, который определит начало профиля носа;
- 6) нос переходит в лоб, причем если отойти от профиля, уменьшая или увеличивая абсциссу x , то наблюдается уменьшение координаты z , но, когда мы доберемся до лба, координата z или не уменьшится, или это уменьшение будет незначительным.

Нахождение начальной точки. Сначала возьмем точку, которая находится посередине рта $(\frac{x_n+x_r}{2}, \frac{y_n+y_r}{2})$. Отметим, что эта точка находится на профиле головы.

Нахождение начала профиля носа. Начнем движение от начальной точки по направлению увеличения ординаты y , проверяя при этом изменение координаты z . Таким образом дойдем до некоторой точки, в окрестности которой начнет резко увеличиваться координата z . Эта точка и определит начало профиля носа.

Нахождение верхушки носа. Верхушку определим, продолжая двигаться в направлении увеличения ординаты y , пока $z'_y > 0$, то есть пока увеличивается координата z . Искомая точка находится по условиям (1) и (2).

$$z'_y(x, y + \lambda) < 0, \quad (1)$$

$$z'_y(x, y - \lambda) > 0. \quad (2)$$

Нахождение конца профиля носа. Далее воспользуемся свойством носа. Начнем исследовать склоны с профилем, которые определяются значениями $z'_x(x - \delta, y), z'_x(x + \delta, y)$. Условие (3):

$$z'_x(x - \delta, y) > z_{x11}, z'_x(x + \delta, y) < z_{x11}, \quad (3)$$

(где z_{x11} – некоторое пороговое значение, которое определяет тангенс угла наклона; $z_{x11} = 0.5$) характеризует наличие склонов. Двигаясь в направлении увеличения ординаты y , доберемся до точки, для которой выполняются условия (4) – (7):

$$z'_x(x - \delta, y) > \varepsilon, \quad (4)$$

$$z'_x(x + \delta, y) < \varepsilon, \quad (5)$$

$$z'_x(x - \delta, y + \lambda) < \varepsilon, \quad (6)$$

$$z'_x(x + \delta, y + \lambda) > \varepsilon. \quad (7)$$

Точка (x, y) будет искомой. Обозначим ее $(x_{нн}, y_{нн})$.

Построение границы носа. На предыдущем шаге был определен профиль носа по склонам на абсциссе x . Теперь, используя точку (x, y) , определим границу носа. Начнем движение от точки $(x_{нн}, y_{нн})$, которая представляет собой точку перехода носа на лоб, вдоль профиля по направлению уменьшения ординаты y с шагом δ . Рассмотрим k -й шаг (8):

$$x_k = x_{нн}, y_k = y_{нн} - k\delta, \quad (8)$$

(x_k, y_k) – точка профиля. Зафиксируем ординату y и начнем с шагом δ уменьшать абсциссу x , наблюдая за величиной $z_x(x, y)$. Движение будет продолжаться, пока $z_x(x, y)$ не станет меньше порогового значения z_{x12} ($z_{x12} = 0.6$). Так найдем некоторую точку $(x_k^{(n)}, y_k^{(n)})$ на левой границе

носа. Совокупность точек $(x_k^{(n)}, y_k^{(n)}), (x_k^{(n)}, y_k^{(n)})$ будет составлять границу носа, а все точки, которые лежат в этой области, формируют сам нос.

Построение линии бровей. Линии бровей начинаются с переносицы. Точка начала линии бровей, была найдена при построении профиля носа и обозначена как точка перехода нос-лоб $(x_{пл}, y_{пл})$. Рассмотрим некоторые особенности бровей. Пусть имеем точку (x, y) на бровях. Это условие запишем так (9):

$$z_x(x, y) < z_{x22}, (z_{x22} = 1.5). \quad (9)$$

Начнем движение от точки (x, y) в направлении уменьшения абсциссы x по линии $y = y_n$. Заметим, что при уменьшении ординаты y на λ , происходит резкое уменьшение ординаты z . Рассмотрим k -й шаг. Имеем точку $(x_k, y_k): x_k = x_{пл} - k\delta, y_k = y_{пл}$. Пусть на k -м шаге имеем точку $x = x_k$. Будем двигаться вдоль нее по точкам $(x_k, y_k + l\lambda), l = 1, 2, \dots$ пока не найдем такую точку $(x_k^{(n)}, y_k^{(n)})$, для которой будет выполняться неравенство (10):

$$z_{yy}^*(x_k, y_k + l\lambda) < z_{y22}. \quad (10)$$

Движение по направлению уменьшения абсциссы x останавливаем, когда выполняется условие $z_x^*(x_k, y_k) > z_{y23} (z_{y23} = -1.3)$, т.е. , когда наблюдается резкое уменьшение координаты z . Таким же способом строим и левую бровь.

Построение контура глаз. Рассмотрим построение правого глаза. Для распознавания глаза используем найденную ранее бровь.

Нахождение крайней правой точки глаза. Начнем движение от средней точки брови по направлению уменьшения абсциссы x с шагом δ . Рассмотрим k -й шаг. Найдем (x_k, y_k) – точку брови. Зафиксируем абсциссу x и начнем двигаться по направлению уменьшения ординаты y (11):

$$x_k = x_{пл} - k\delta, \quad (11)$$

Точку контура глаза найдем из условий (12) – (14):

$$z_{yy}^*(x_k, y_{kl1}) > z_{y24}, \quad (12)$$

$$z_{yy}^*(x_k, y_{kl2}) < 0, l_1 < l_2 < l_3, \quad (13)$$

$$z_{yy}^*(x_k, y_{kl3}) > z_{y24}, \quad (14)$$

где z_{y24} – некоторое пороговое значение, которое определяется на практике. Эти условия определяют наличие вогнутости на краях глаза и выпуклости посередине. Если возникает ситуация $l_3 - l_2 < 2$, то это неравенство определит крайнюю правую точку глаза.

Обратное движение. Движение организуем аналогично предыдущему способу, двигаться по брови будем справа налево, пока не найдем левую точку глаза.

Так же будет строиться контур левого глаза.

Построение контура губ. Построение линии, которая проходит между губами. Движение начнем с крайней правой точки губ (x_n, y_n) по направлению увеличения абсциссы x с шагом δ (15).

$$x_{k(г)} = x_n + k\delta. \quad (15)$$

Величину $y_{k(г)}$ определим следующим образом (16):

$$y_{k(г)} = y_{k(г)} + l\lambda, \text{ где } l = \operatorname{argmin}_{l=-1,1} z_{yy}^*(x_{k(г)}, y_{k-1(г)} + l\lambda). \quad (16)$$

Построение линии, которая ограничивает верхнюю губу. Движение начнем с крайней правой точки губ (x_n, y_n) по направлению увеличения абсциссы x с шагом δ .

$$x_{k(\text{в})} = x_n + k\delta. \quad (17)$$

Величину $y_{k(\text{в})}$ определим следующим образом:

$$y_{k(\text{в})} = y_{k(\text{с})} + l\lambda, \text{ где } l = \underset{\substack{l \in \mathbb{Z} \\ l > 0}}{\text{arg max}} z_{yy}''(x_k, y_{k(\text{с})} + l\lambda). \quad (18)$$

Построение линии, которая ограничивает нижнюю губу. Эта линия строится аналогично предыдущей и обозначается l :

$$l = \underset{\substack{l \in \mathbb{Z} \\ l < 0}}{\text{arg max}} z_{yy}''(x_k, y_{k(\text{в})} + l\lambda) \quad (19)$$

Выводы

В работе разработан и реализован алгоритм, который позволяет решить задачу моделирования и распознавания мимических проявлений эмоциональных состояний. Основная идея предложенной технологии состоит в алгоритме автоматического получения гибких шаблонов (на базе NURBS-кривых) контуров носа, бровей, глаз и рта.

Предложенные методы способствуют автоматическому нахождению контуров основных частей лица, определяя эмоции на изображении, преобразовывая B-сплайн кривые в контуры лица человека.

Данная технология имеет практическую ценность в системах визуального контроля эмоционального состояния человека во многих сферах деятельности, где требуется большая сосредоточенность и необходимость принимать решения в режиме реального времени.

Литература

1. Кривонос Ю.Г. Моделирование и анализ мимических проявлений эмоций / Ю.Г. Кривонос, Ю.В. Крак, А.В. Бармак // Доклады НАНУ, 2011. - №12. - С. 51-55.
2. Ekman P. Learning to Make Facial Expressions / P. Ekman, W.V. Friesesen. – Part II. – Palo Alto, 2009. Режим доступа: http://mplab.ucsd.edu/wpcontent/uploads/wu_icdl20091.pdf
3. Ekman P., Facial Action Coding System. / P. Ekman, W.V. Friesesen. – Part II. – Palo Alto, 2006. Режим доступа: <https://pdfs.semanticscholar.org/99bf/8ac8c131291d771923d861b188510194615e.pdf>
4. Леонтьев В.О. Классификация эмоций / В.О. Леонтьев. – Одесса: ИИЦ, 2006. – 84 с.
5. Ефимов А.Н. Моделирование и распознавание мимических проявлений эмоций на лице человека / Г.М. Ефимов // Искусственный интеллект, 2009. - С. 532-542.6. Крак Ю.В. Синтез мимических выражений эмоций на основе формальной модели / Ю.В. Крак, А.В. Бармак, М. Ефимов // Искусственный интеллект, 2007. - №2 - С. 22-31.
7. Piegl L. The NURBS Book / Les Piegl, Wayne Tiller. – [2nd Edition]. – Berlin : Springer-Verlag, 1996. – 646 p.
8. Ekman P. Cross-cultural studies of facial expression / P. Ekman, W.V. Friesesen // Darwin and facial expression: A century of research in review. – N.Y : Academic, 2010. – P. 196-222.
9. Изард К. Эмоции человека / К. Изард. – М. : Изд-во МГУ, 2010. – 439 с.

УДК 004.55

СОЗДАНИЕ WEB-САЙТА ДЛЯ МАГАЗИНА СТРОИТЕЛЬНЫХ МАТЕРИАЛОВ

Титов К.В., Радевич Е.В.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа
E-mail: kirya898@mail.ru; [radevich katerina@mail.ru](mailto:radevich_katerina@mail.ru)

Аннотация

Титов К.В., Радевич Е.В. Создание Web-сайта для магазина строительных материалов. Проанализированы современные виды Web- сайты. Рассмотрены основные технологии, тенденции и перспективы развития сайта.

Ключевые слова: разработка адаптивного сайта, с использованием современного языка программирования, с базой данных и встроенным искусственным интеллектом.

Annotation

Titov K. V., Radevich E. V. Creation of Web-site for building materials store. Modern types of Web - sites are analyzed. The main technologies, trends and prospects of the site development are considered.

Keywords: development of an adaptive site, using a modern programming language, with a database and built-in artificial intelligence.

Общая постановка проблемы

Web-сайт должен обладать следующими особенностями: гибкостью и простой системой управления. Также должна быть реализована возможность печати web-страницы и возможность обратной связи для получения ответа на интересующие вопросы в кратчайшие сроки. Отметим, что Web-сайт должен быть разработан с максимальной простотой использования пользователями.[1]

Анализ принципов построения Web-сайта

Удачный Web-сайт – это в высшей степени эффективный инструмент торговли - он способен завлечь потенциальных покупателей. Как и любой другой маркетинговый инструмент, основанный на принципе непосредственного отклика, прежде всего, он должен заинтриговать посетителя, а затем сподвигнуть его на определенные действия. Однако, многие игнорируют эту особенность главной страницы, что часто приводит к тому, что посетители не задерживаются на сайте надолго и покидают его, едва зайдя. Такие Web-сайты, пусть даже содержащие иногда огромное количество полезных советов и статей, практически никогда не достигают предполагаемого уровня посещаемости, не говоря уже о продажах. Сделав всего несколько изменений, простой Web-сайт может превратиться в более надежный и эффективный инструмент. Важно помнить, что изо дня в день на потенциальных клиентов обрушивается поток информации и различных рекламных сообщений, и что в плане завоевания их внимания существует предельно жесткая конкуренция. Web-сайт, способный привлечь внимание и вызвать любопытство, побудит клиентов не только просмотреть оставшиеся страницы и совершить покупки, но и снова посетить его через некоторое время, а также рекомендовать своим друзьям и знакомым.[2]

Обоснование и выбор программного обеспечения

Для создания Web-сайта была выбрана программа Sublime Text 3, т.к. ее гибкие функциональные возможности позволяют создавать более совершенные Web-узлы, включающие средства для профессионального проектирования, разработки, работы с данными и публикации, необходимые для создания динамических и более сложных Web-

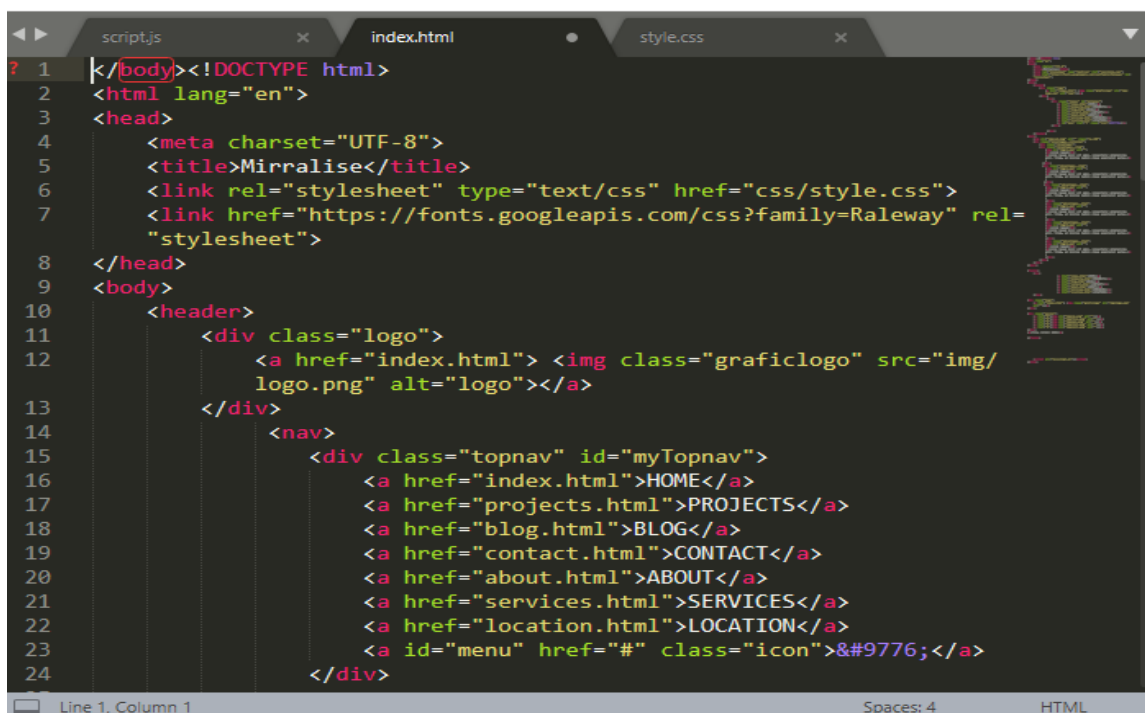
узлов. Sublime Text 3 позволяет усовершенствовать процесс Web-разработки в следующих трех ключевых областях:

1. Проектирование – усовершенствованные средства проектирования позволяют улучшить оформление Web-узлов. Новые средства разметки и работы с графикой упрощают процесс создания Web-узлов, полностью отвечающих замыслу пользователя.

2. Написание кода – средства проектирования позволяют повысить качество создаваемого кода и усовершенствовать навыки в области программирования. Встроенные средства разработки сценариев обеспечат поддержку интерактивности в создаваемых продуктах. Благодаря профессиональным средствам написания кода можно работать быстрее, эффективнее и точнее. [3]

3. Расширение возможностей – можно организовать общение и обмениваться данными по-новому, создавая управляемые данными Web-узлы с широкими интерактивными возможностями в редакторе, действующем по принципу – «что видишь на экране, то и получишь при печати». Усовершенствованные функциональные возможности публикации ускорят размещение создаваемых Web-страниц в Интернете.

Средства управления упрощают работу с несколькими расположенными рядом изображениями и фрагментами и позволяют создавать эффекты отображения, например, всплывающие меню. Ориентация на Web-обозреватель или разрешение экрана происходит благодаря возможности распознавания. Можно также увидеть как Web-узел будет выглядеть при различных комбинациях Web-обозревателей и разрешений. Создание таблиц, используемых для разметки, и работа с ними, а также контроль разметки с точностью до пикселя приведен на рисунке 1.



```
? 1 |</body><!DOCTYPE html>
2 |<html lang="en">
3 |<head>
4 |  <meta charset="UTF-8">
5 |  <title>Mirralise</title>
6 |  <link rel="stylesheet" type="text/css" href="css/style.css">
7 |  <link href="https://fonts.googleapis.com/css?family=Raleway" rel="
8 |  "stylesheet">
9 |</head>
10|<body>
11|  <header>
12|    <div class="logo">
13|      <a href="index.html"> </a>
15|    </div>
16|    <nav>
17|      <div class="topnav" id="myTopnav">
18|        <a href="index.html">HOME</a>
19|        <a href="projects.html">PROJECTS</a>
20|        <a href="blog.html">BLOG</a>
21|        <a href="contact.html">CONTACT</a>
22|        <a href="about.html">ABOUT</a>
23|        <a href="services.html">SERVICES</a>
24|        <a href="location.html">LOCATION</a>
25|        <a id="menu" href="#" class="icon">&#9776;</a>
26|      </div>
```

Рисунок 1 – Процесс создания Web-страницы с помощью Sublime Text

Ускорение и упрощение генерации кода

Средства проектирования, предлагаемые Sublime Text 3, позволяют создавать эффективные чистые HTML-файлы и могут применяться как профессионалами, так и новичками в области Web-разработки. Пользуясь преимуществами генерируемого с помощью Sublime Text 3 эталонного кода, профессионалы могут применять

усовершенствованные средства написания кода и средства работы со сценариями для обеспечения интерактивных возможностей работы с Интернетом. Благодаря простоте средств написания кода с их помощью можно начать освоение HTML.[4]

Средства написания кода

Представление с разделением позволяет увидеть изменения, вносимые в представление конструктора, обеспечивают обновление представления кода. Простота выбора и изменения тегов, а также управления ими с помощью быстрого выбора тегов и быстрого редактора тегов. Технология Microsoft IntelliSense® упрощает написание кода и снижает вероятность ошибок. Эта технология доступна для HTML (Hypertext Markup Language), таблиц каскадных стилей, XSLT (Extensible Stylesheet Language), Microsoft JScript® и Microsoft ASP.net. Удаление лишнего кода, генерируемого Word 2010 или другими программами Web-разработки.[5]

Выводы

В заключении отметим, что на сегодняшний день создать свой интернет-проект можно множеством способов. Самый сложный вариант – это самостоятельно все разработать.

Выделим следующие этапы разработки сайта самостоятельно:

- 1) проектирование;
- 2) прорисовка макета в графическом редакторе;
- 3) написание программы;
- 4) тестирование;
- 5) управление.

Плюсы данного способа в том, что на выходе можно получить очень качественный уникальный ресурс.

Другой способ заключается в том, что сначала ставим любую популярную cms (чаще всего wordpress) и там уже есть тысячи готовых бесплатных шаблонов. Минус только в том, что поскольку шаблон бесплатно распространяется, то данное оформление не будет уникально.[6]

Литература

1. Байтов.В Профессиональная разработка сайтов на Drupal 7 / В. Байтов – М.: Вильямс, 2012. – 304 с.
2. Венедюхин. А. Создание сайтов / А.Венедюхин – М.: Вильямс, 2014. – 304 с.
3. Гардер. А. Разработка веб-сайтов / А.Гардер – М.: Вильямс, 2011. – 205 с.
4. Дронов. В. HTML и CSS / В.Дронов – М.: Вильямс, 2012. – 150 с.
5. Дронов. В. HTML 5 и MySQL 5/6 / В.Дронов – М.: Вильямс, 2009. – 201 с.
6. Костин С.П. Самоучитель создание Web-сайтов / С.П.Костин – М.: Триумф, 2009. – 176 с.

УДК 004.93

МЕТОДИКА РАСПОЗНАВАНИЯ ПЕЧАТНОГО ТЕКСТА С ВЫДЕЛЕНИЕМ ГРАФИЧЕСКОГО МАТЕРИАЛА

Шумский А.А., Кравец Т.Н.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: shumskiyalexander@gmail.com t.n.kravets@gmail.com

Аннотация:

Шумский А.А., Кравец Т.Н. Методика распознавания печатного текста с выделением графического материала. Проведен анализ актуальности рассматриваемой темы. Рассмотрены этапы предобработки изображения для дальнейшего распознавания. Приведена методология выделения текста и графического материала на исходном изображении.

Annotation:

Shumskiy A.A., Kravets T.N. The method of recognition of printed text with the selection of graphic material. The analysis of the relevance of the topic. The stages of image preprocessing for further recognition are considered. The methodology for the selection of text and graphic material on the original image.

Введение

В век информационных технологий и электронных подписей обычные бумажные документы ничуть не утратили своей силы. Работникам различных сфер деятельности зачастую приходится вручную набирать объемные тексты из-за отсутствия или невозможности получить в свое распоряжение электронные исходники. Сам текст может быть наполнен различными графическими материалами, что усложняет этот процесс. Для его автоматизации существуют системы распознавания печатного текста с выделением графического материала. Данные системы бывают в виде сервисов и приложений под разные платформы (Windows, Android и другие) [1].

Актуальность проблемы

Сейчас существует множество систем для распознавания текста, однако лишь малая их часть способна выделять графический материал и переносить его в итоговый электронный документ с сохранением его положения.

Для этого существует множество методов сегментации исходного изображения. Каждый эффективен в разных случаях. Задача выделения графического материала сильно усложняет процесс, поскольку распознавание не должно срабатывать на графическом материале. Также требуется минимизировать операции для улучшения быстродействия итогового программного продукта.

Цели и задачи исследования

Целью исследования является анализ существующих методов распознавания текста и выделение графического материала на исходном изображении, а также создания собственного программного продукта на их основе.

Среди основных задач исследования по распознаванию печатного текста с выделением графического материала можно выделить:

- выбор фильтров предобработки исходного изображения;
- изучение методов сегментации текста и графического материала;
- формирование модели нейронной сети для распознавания;
- выполнение проверки и вывод результата в форматированный файл.

Распознавание текста и выделение графического материала

Распознавание печатного текста является довольно сложной задачей. Человеку это дается просто, но машине требуется много подготовительных действий. И только после них происходит распознавание.

В теории распознавания образов [2] можно выделить два основных направления:

– изучение способностей к распознаванию, которыми обладают существа, объяснение и моделирование их;

– развитие теории и методов построения устройств, предназначенных для решения отдельных задач в прикладных целях.

Среди основных подходов к распознаванию текста можно выделить три основных метода:

– сравнение с шаблоном;

– распознавание по критериям;

– распознавание при помощи самообучающихся алгоритмов.

Последний из методов подразумевает использование нейронных сетей в том или ином виде. Именно он является наиболее качественным среди всех известных методов, однако наиболее ресурсоемким.

Для максимально эффективного распознавания текста необходимо выполнить действия по предобработке и постобработке. Работу программного средства распознавания печатного текста можно поделить на несколько этапов (рис. 1).

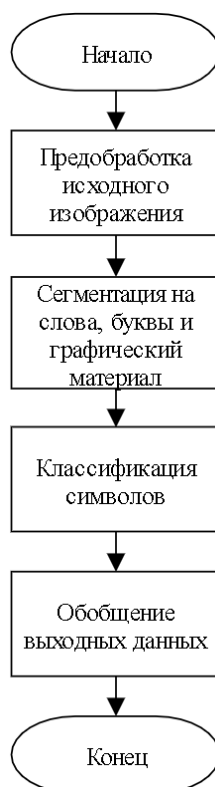


Рис. 1. Обобщенные этапы алгоритма

Входными данными являются фотографии исходного текста. Для повышения точности распознавания необходимо провести предобработку изображения. Далее обработанное изображение поступает на модуль сегментации, где выделяются буквы и графические элементы. Данные буквы классифицируются с помощью нейронной сети, а

затем идет поиск и сравнение итоговых слов со словарем. В результате в итоговый документ с форматированием вносится распознанный текст и найденные изображения.

Предобработка исходного изображения

На этапе предобработки необходимо улучшить изображение, применив различные фильтры.

Обычно текст использует один цвет. Применим классический черно-белый фильтр чтобы подготовить исходное изображение к распознаванию.

Для удаления шумов используется медианный фильтр. Значения отсчетов внутри окна фильтра сортируются в порядке возрастания (убывания); и значение, находящееся в середине упорядоченного списка, поступает на выход фильтра. В случае чётного числа отсчетов в окне выходное значение фильтра равно среднему значению двух отсчетов в середине упорядоченного списка. Окно перемещается вдоль фильтруемого сигнала и вычисления повторяются.

Для выделения самого текста приведем изображение к бинарному (монохромному) виду. Границу перехода в фильтре определим как среднее значение яркости исходного изображения. Такой переход позволит значительно ускорить работу дальнейших алгоритмов.

После данных преобразований на изображении четко виден текст. Получившийся файл уже можно сегментировать на блоки.

Выделение текста и графики

На данном этапе необходимо выделить графические изображения на холсте, а также блоки с текстом. Для этого используем на монохромном изображении фильтр «размазывания», который для каждой яркой точки закрашивает соседние пиксели. Порог мощности определяется исходя из разрешения исходного файла. Таким образом буквы и слова наползают друг на друга, превращаясь в единый объект и сформируют блоки, в которых находится или графическое изображение, или текст.

Определим границы блоков на изображении для дальнейшей работы. Воспользуемся классическим алгоритмом «волшебной палочки». В начале случайным образом выбирается закрашенный пиксель на монохромном размазанном изображении. Затем в окрестностях выбранной точки закрашиваются все пиксели с таким же цветом пока не останется закрашенных точек у исходного объекта. Найдем закрашенные крайние точки на оси абсцисс и ординат рассматриваемого объекта в блоке. Это и будут границы на исходном изображении.

По закрашенной матрице «магической палочки» скопируем объект для распознавания из исходного монохромного файла, а также сохраним в памяти исходные координаты. Таким образом мы найдем все блоки для распознавания, и они не будут пересекаться.

Для найденных блоков необходимо определить тип (изображение или текст). Это делает пользователь через интерфейс программы. Если в процессе дальнейшей сегментации текста не будут найдены слова и строки в обрабатываемом блоке, то тогда этот блок автоматически меняет свой тип на изображение.

Для дальнейшего распознавания необходимо выделить в каждом текстовом блоке строки, слова и символы.

Поиск строк в обработанном блоке происходит на основе сравнения среднего значения яркости в горизонтальных пиксельных линиях. Соответственно строка текста будет находиться там, где ее яркость будет заметно выше нуля.

Поиск слов похож на поиск строк. Воспользуемся фильтром размытия небольшой мощности с целью сплавить буквы в единый объект, но оставить интервал между словами. После этого проведем поиск границ слов на основе оценки среднего значения яркости вертикальных пиксельных линий. Соответственно слово будет находиться там, где ее яркость будет заметно выше нуля.

Поиск символов в слове отличается от поиска строк и слов, поскольку буквы иногда могут наплывать друг на друга. Для поиска символов в слове необходимо сначала составить таблицу локальных минимумов яркости. Затем нужно удалить ложные разделители. Для этого удаляем индексы столбцов пикселей, средняя яркость которых в окрестностях справа и слева выше граничного значения, определенного экспериментально для каждого языка и шрифта. Для увеличения точности можно проделать предыдущую операцию отдельно для верха буквы, середины и низа. Это особенно актуально для таких букв, как «Ц», «Н», «П» и т.д.

Результатом операции сегментирования является символ, привязанный к соответствующим объектам слов, строк и блоков, готовый к распознаванию нейронной сетью.

Распознавание текста

Искусственные нейронные сети достаточно широко используются при распознавании символов. Алгоритмы, использующие нейронные сети для распознавания символов, часто строятся следующим образом. Поступающее на распознавание изображение символа (растр) приводится к некоторому стандартному размеру. Как правило, используется растр размером 16x16 пикселей. Значения яркости в узлах нормализованного растра используются в качестве входных параметров нейронной сети. Число выходных параметров нейронной сети равняется числу распознаваемых символов. Результатом распознавания является символ, которому соответствует наибольшее из значений выходного вектора нейронной сети. Повышение надежности таких алгоритмов связано, как правило, либо с поиском более информативных входных признаков, либо с усложнением структуры нейронной сети [3].

Для реализации распознавания была выбрана двухслойная нейронная сеть. На первом слое происходит сравнение по значению яркости исходного символа. На втором слое происходит сравнение по контурам.

Обобщение выходных данных

Результаты работы комплекса распознавания текста с выделением графического материала будут выводиться и сохраняться в виде форматированного текста. Предполагается их размещение в рамки по координатам исходных блоков. Текст будет вписан в рамку с учетом подходящего размера шрифта. Графические материалы будут перенесены с исходного изображения по соответствующим координатам.

Для увеличения точности распознавания текста пользователь может использовать функцию сравнения слов с обширным словарем в автоматическом или ручном режиме.

Литература

1. Шумский А.А. Программное средство распознавания печатного текста / А.А.Шумский, Е.В.Бычкова // ИУСМКМ – Донецк : Сборник материалов VIII Международной научно-технической конференции в рамках III Международного Научного форума ДНР, 2017. – С. 402.
2. Теория распознавания образов [Electronic resource] / Интернет-ресурс. - Режим доступа : [www/ URL: http://ru.wikipedia.org/wiki/Теория_распознавания_образов](http://ru.wikipedia.org/wiki/Теория_распознавания_образов). - Загл. с экрана.
3. Мисюрёв А.В. Использование искусственных нейронных сетей для распознавания рукопечатных символов [Electronic resource] / Интернет-ресурс. - Режим доступа : [www/ URL: http://ocrai.narod.ru/hp.html](http://ocrai.narod.ru/hp.html). - Загл. с экрана.

УДК 004.6, 004.91

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ПРЕДПРИЯТИЯ ОПТОВОЙ ТОРГОВЛИ С ИСПОЛЬЗОВАНИЕМ ПРОГРАММЫ «1С:ПРЕДПРИЯТИЕ» ВЕРСИИ 8.2

Ярмак М.А, Семенова А.П.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: mansfik1@gmail.com, nastena-semenova19@rambler.ru

Аннотация

Ярмак М.А, Семенова А.П. Проектирование и разработка информационной системы предприятия оптовой торговли с использованием программы «1С:Предприятие» версии 8.2. Работа посвящена разработке базы данных в системе 1С:Предприятие 8.2 для автоматизации работы оптово торгового предприятия. Приведены основные требования к функционалу системы, описаны составные части разработанной базы данных.

Ключевые слова: автоматизация оптовой торговли, модуль планирования, модуль управления, управление запасами, быстродействие системы.

Annotation

Yarmak M.A., Semenova A.P. Database engineering and development for enterprise wholesale trade using the program «1С:Предприятие» version 8.2. This work is dedicated to developing database using «1С:Предприятие» system for automation work on wholesale trade enterprise. The main requirements for functional system are given, components of the developed database are described.

Key words: automation work on wholesale trade enterprise, sales planning module, procurement management, stocks management, speed performance for the system.

Введение. Письменный процесс заполнения документов является достаточно трудоемким и отнимает слишком много времени у пользователя, так же как и планирование продаж или управление заказами покупателей, закупками, запасами, накладными расходами. Поэтому требуется база данных, которая может быть гибкой, что позволит проводить ее расширение и доработки, если возникнет необходимость увеличить функциональные возможности. База должна иметь мощные средства взаимосвязанного планирования и контроля продаж и закупок что позволит комплексно решать задачи ведения заказов покупателей, резервирование товаров на складах, выявление текущих и перспективных потребностей в товарах, формирование на их основе заказов поставщикам [1].

Цель статьи. Изучение особенностей разработки баз данных, модулей и документов, а так же разработка базы данных для оптово-торгового предприятия в системе «1С:Предприятие» 8.2.

Постановка задачи. В первую очередь необходимо реализовать планирование продаж на предприятии. Реализованные в системе возможности планирования и план-фактного анализа обеспечивают поддержку принятия решений, определяющих развитие бизнеса, и позволяют перейти от «стихийного» управления к регулярному менеджменту продаж.

Система дает управленческому персоналу компании мощные и удобные инструменты для планирования объемов продаж в натуральном и стоимостном выражении, в том числе на основании данных о продажах за предыдущие периоды,

Во-вторых, требуется создать управление заказами покупателей, реализация, которой позволит оптимальным образом размещать заказы покупателей в заказах поставщикам и резервировать товары на складах в соответствии с принятой в компании стратегией исполнения заказов и схемами работы.

Все этапы прохождения заказа и его корректировки фиксируются в системе соответствующими документами, что позволяет в любой момент получить полную информацию о ходе выполнения заказа, отслеживать историю взаимоотношений с клиентами и поставщиками, оценивать эффективность и надежность работы с ними.

В третьих требуется разработать управление закупками. 1С Предприятие обеспечивает менеджеров торгового предприятия информацией, необходимой для своевременного принятия решений о пополнении товарных запасов, для снижения затрат на закупки и четкой организации взаимодействия с поставщиками. Основными функциями подсистемы управления закупками являются, оперативное планирование закупок, оформление заказов поставщикам и контроль их исполнения и платежный календарь расхода денежных средств.

В четвертых нужно реализовать управление запасами на предприятии, что позволит вести детальный оперативный учет товаров на складах, обеспечивается полный контроль товарных запасов предприятия в оптовой и розничной торговле. Все складские операции (поступление товаров, перемещение товаров, реализация товаров, инвентаризация и т.п.) фиксируются с помощью соответствующих документов.

Это позволяет вести учет товаров в различных единицах измерения (упаковках), складской учет на множестве складов (мест хранения). Таким образом, поддерживается раздельный учет собственных товаров, товаров, принятых и переданных на реализацию, товаров, принятых на ответственное хранение, а также учет возвратной тары.

В пятых нужно ввести в программу учёт накладных расходов таких как, затрат на транспортировку, хранение, оплаты пошлин и акцизов, услуг как собственных, так и сторонних организаций и т.д. Эти расходы могут быть отнесены на себестоимость товаров. В зависимости от характера накладных расходов, их распределение по товарным позициям может производиться пропорционально стоимости поступивших товаров, их весу или количеству.

Исследования. Программное обеспечение реализовано с целью уменьшения времени пользователя на заполнение документов, таблиц, планирование принятия решений, планирование объемов продаж, управление заказами покупателей, управление закупками, управление запасами на предприятии и учета накладных расходов.

В ходе разработки программного обеспечения были исследованы следующие понятия:

- виды отчетов в 1С:Предприятие 8.2 [2];
- архитектура баз данных в 1С:Предприятие 8.2 [3];
- поиск наиболее лучшего быстродействия системы;
- виды регистров и их значимость при работе с данными [4];
- анализ рассмотренных аналогов программного обеспечения.

В процессе исследования предметной области был выявлен недостаток скорости при расчетах в таблицах и проведении документов. Кроме этого было определено, что регистры сведений, накопления, бухгалтерии и расчета играют важную роль. Все регистры являются важной частью системы, и для их исправной работы требуется правильная связанность с документами, отчетами и таблицами. Регистры сведений и накопления предназначены для хранения информации, развернутой по комбинации измерений. Единственное их отличие в том, что регистр накопления может хранить только данные поддающиеся суммированию.

Программная реализация. Программное обеспечение предоставляет пользователю весь необходимый функционал для наиболее удобной работы на оптово-торговом

предприятия. Программный продукт представлен в виде базы в системе «1С:Предприятие»
8.2. Сама база состоять из следующих конструкторов (модулей):

- подсистемы (налоговый учет, оборотные активы, расчет зарплаты, аналитическая информация, управление денежными средствами, управление персоналом и т.д.);
- общие модули (бухгалтерских отчетов, заполнения документов, проведение расчетов, процедура обмена данными, типовые отчеты, типовые операции и т.д.);
- общие формы (печать документов, результат проведения, форма выбора сотрудников, склада, режима заполнения, печатных форма, форма описания счета и т.д.);
- документы (авансовый отчет, внутренний заказ, выработка ОС, НМА, договоренность, закрытие заказов, инвентаризация ОС, НМА, отчет о розничных продажах, производства за смену и т.д.);
- отчеты (анализ заказа, валовая прибыль, ведомость товаров на складе, ведомость товары организации, закупки, затраты, кадровый план, кассовая книга и т.д.);
- планы видов расчета (взносы в фонды, начисления, отражение зарплаты организации, средний заработок, удержания);
- регистры сведений (время действия скидок, кадровый план, категории объектов, значения точки заказа, курсы валют, лимит возвратной тары и т.д.);
- регистры накоплений (брак в производстве, внутренние заказы, выработка НМА, выработка ОС и т.д.);
- регистры бухгалтерии (налоговый и хозяйственный);
- регистры расчета (взносы в фонд, начисления, отражение зарплаты в учетах, расчет среднего заработка, удержания).

На рисунке 1 приведен пример окна и журнал складские документы.

Дата	Номер	Вид документа	Вид операции	Сумма	Валюта	Контрагент	Склад
01.01.2011 0:00:00	ТТ000000001	Оприходование тов...	Товары, прод...	153 607,55			Участок 1
01.01.2011 9:00:00	Гм000000002	Поступление товар...	Покупка, ком...	110 039,42	грн	Полиграфиче...	Основной
06.01.2011 12:00:00	Гм000000055	Реализация товаро...	Продажа, ком...	0,02	грн	Альтер Медиа...	Основной
06.01.2011 12:00:00	Гм000000056	Реализация товаро...	Продажа, ком...	0,02	грн	Альтер Медиа...	Основной
06.01.2011 12:00:00	Гм000000057	Реализация товаро...	Продажа, ком...	0,02	грн	Альтер Медиа...	Основной
06.01.2011 12:00:02	Гм000000054	Реализация товаро...	Продажа, ком...	0,02	грн	Альтер Медиа...	Основной
06.01.2011 12:00:02	Гм000000002	Реализация товаро...	Продажа, ком...	23 479,72	грн	Укрпочта" Д...	Основной
06.01.2011 12:00:03	Гм000000003	Реализация товаро...	Продажа, ком...	8 217,30	грн	Укрпочта" Д...	Основной
06.01.2011 12:00:04	Гм000000004	Реализация товаро...	Продажа, ком...	1 200,50	грн	Новый Мир" ...	Основной
06.01.2011 21:00:01	Гм000000001	Реализация товаро...	Продажа, ком...	17 606,90	грн	Укрпочта" Д...	Основной
07.01.2011 12:00:00	Гм000000008	Поступление товар...	Покупка, ком...	16 500,00	грн	Альтер Медиа...	Основной
12.01.2011 12:00:01	Гм000000042	Реализация товаро...	Продажа, ком...	480,00	грн	Донбас-креди...	Основной
13.01.2011 12:00:01	Гм000000005	Реализация товаро...	Продажа, ком...	1 210,30	грн	Новый Мир" ...	Основной
13.01.2011 12:00:02	Гм000000006	Реализация товаро...	Продажа, ком...	17 606,90	грн	Укрпочта" Д...	Основной
13.01.2011 12:00:03	Гм000000007	Реализация товаро...	Продажа, ком...	23 479,72	грн	Укрпочта" Д...	Основной
13.01.2011 12:00:04	Гм000000008	Реализация товаро...	Продажа, ком...	8 475,30	грн	Укрпочта" Д...	Основной
14.01.2011 12:00:00	Гм000000007	Поступление товар...	Покупка, ком...	17 050,00	грн	Альтер Медиа...	Основной
17.01.2011 12:00:00	Гм000000018	Поступление товар...	Покупка, ком...	11 751,00	грн	Почтампл шлз ...	Основной

Рисунок 1 – Пример окна и журнал складские документы

В стартовой форме для удобства пользователя доступны все нужные ему конструкторы и формы для работы (операции, справочники, документы, отчеты, сервис и т.д.). Приложение строится таким образом, что бы пользователю были доступны все необходимые инструменты для работы на предприятии. Так же система «1С:Предприятие» обладает свойством, позволяющем дорабатывать базу не отрывая пользователя от работы. Для этого

ему требуется просто обновить системы после ее доработки. В данной базе вся работа сфокусирована на функциональности программного обеспечения, никаких визуальных наработок в ней нет.

Выводы. В ходе исследования и разработки программного продукта были определены цели и задачи, необходимые для наилучшего функционирования и быстродействия базы данных в системе «1С:Предприятие» 8.2. Рассмотрены модули и конструкты системы, общие формы, документы, отчеты и регистры, требуемые для осуществления потребностей предприятия и его сотрудников.

Литература

1. Радченко М.Г. 1С:Предприятие 8.1. Практическое пособие разработчика. Примеры и типовые приёмы. / М.Г. Радченко – СПб: Питер, 2007. – 413с.
2. Митичкин С.А. Разработка в системе 1С:Предприятие 8.0. / С.А. Митичкин – СПб: Питер, 2003. – 523с.
3. Радченко М.Г. Архитектура и работа с данными 1С:Предприятие 8.2. / М.Г. Радченко – СПб: Питер, 2010. – 370с.
4. Хрусталева Е.Ю. Разработка сложных отчетов в 1С:Предприятии 8. Система компоновки данных. / Е.Ю.. Хрусталева – СПб: Питер, 2001. – 452 с.

УДК 004.91:004.738.52

РАЗРАБОТКА ЭКСПЕРТНОЙ СИСТЕМЫ ДЛЯ ПРОВЕРКИ ВЕБ-САЙТОВ В КАЧЕСТВЕ SEO ТЕХНОЛОГИЙ

Деньгуб Ю.О., Гудаев О.А.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: yulechka-yuliya-99@mail.ua

Аннотация:

Деньгуб Ю.О., Гудаев О.А. Разработка экспертной системы для проверки веб-сайтов в качестве SEO технологий. Рассматривается создание экспертной системы проверки сайта на соответствие SEO оптимизации. Рассмотрена архитектура клиентского приложения реализующего экспертную систему.

Annotation:

Dengyb Y.O., Gydaev O.A. Development of an expert system for checking websites as SEO technologies. The article to consider the creation of an expert system to check the site for compliance with SEO optimization. The architecture of the client application implementing the expert system is considered.

Общая постановка проблемы

Выявить основные правила SEO оптимизации, применимые в алгоритме проверки веб-сайта. Разработать архитектуру приложения, реализующую алгоритм проверки.

Исследование логической модели. Согласно технологии SEO, оптимальный сайт – это степень удобства сайта для пользователя [1]. Это комплексное понятие, которое включает цветовое оформление страниц, расположение информационных блоков и меню, типы и размеры шрифтов, удобство навигации по сайту и взаимодействия с элементами интерфейса [2]. Для исследованной в работе [3] модели использования онтологии в UX-дизайне графического интерфейса сайта можно предложить структуру онтологий SEO оптимизации сайта показанную на рисунке 1.

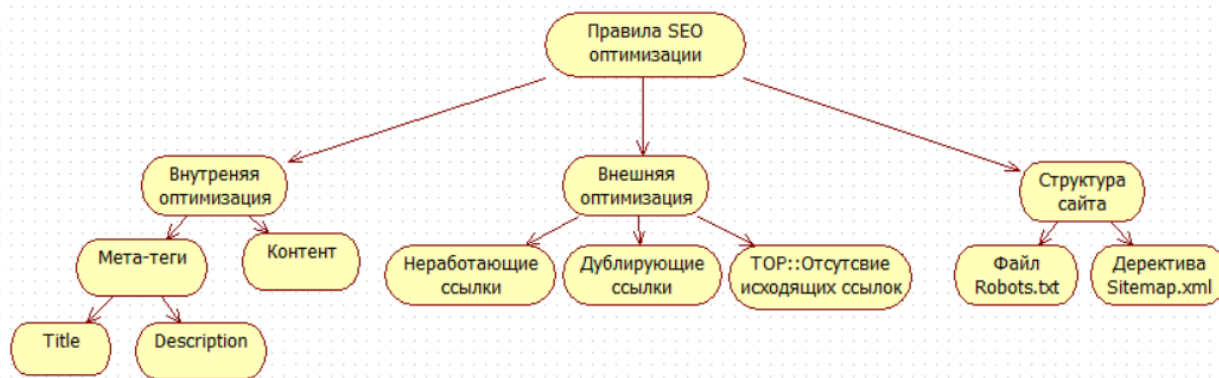


Рисунок 1 – Структура SEO оптимизации сайта

Логическая модель экспертной системы проверки сайта на соответствие SEO оптимизации содержит ряд правил. Правило №1. Проверка на наличие мета-тегов заголовка

и описания (Title, Description) и достаточного количества символов в них. Не повторяемость текста тегов на различных страничках сайта. Правило №2. Проверка на наличие и достаточного количество символов размещенного контента на главной странице. Правило №3. Проверка на наличие файла sitemap.xml в каталоге сайта. Файл обязательно должны содержать URL-адреса всех страниц сайта. Правило №4. Проверка сайта на наличие файла robots.txt, а также присутствие в нем директив Disallow, Host. Правило №5. Составное правило для частей контента сайта. Корректно обрабатывать внутренние и исходящие ссылки: отсутствие неработающих ссылок; отсутствие дублирующих ссылки на целевых страницах; отсутствие исходящих ссылок на сторонние сайты.

Для реализации экспертной системы спроектирована архитектура компонентов приложения, приведенная на рисунке 2.

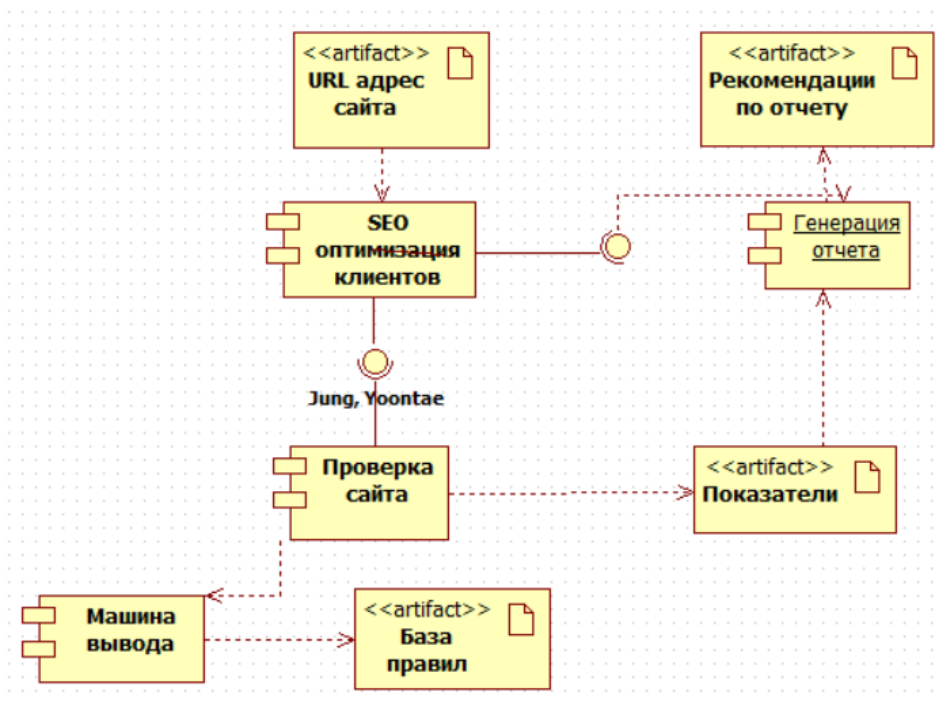


Рисунок 2 – Архитектура коипонентов приложения

Выводы

Выявлены основные правила автоматического исследования сайта на оптимальность, применимые в алгоритме проверки экспертной системой. Разработана архитектура приложения проверки сайта на соответствие SEO оптимизации.

Литература

1. Граппоне, Дж. Поисковая оптимизация сайтов. Исчерпывающее руководство / Дженнифер Граппоне, Градива Казн ; [пер. с англ.]. – М. : Эксмо, 2012. – 528 с. : ил.
2. Рябов, В.А. Современные веб-технологии / В.А. Рябов, А.И. Несвижский. – [2-е изд., испр.] – М. : Национальный Открытый Университет ИНТУИТ, 2016. – 1008 с.
3. Деньгуб, Ю. О. Онтология в UX-дизайне графического интерфейса сайта / Ю. О. Деньгуб, О.А. Гудаев // Материалы студенческой секции IX Международной научно-технической конференции «Информатика, управляющие системы, математическое и компьютерное моделирование» (ИУСМКМ-2018), 22-24 мая 2018 г. – Донецк : ДонНТУ, 2018. – С. 290-292.

УДК 004.85

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИЙ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ПРИ ИГРЕ В ПОКЕР

Котлубеев А.Ю., Головченко А.А., Бочаров Д.М.
Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа
E-mail: uasasharpg@gmail.com

Аннотация:

Котлубеев А.Ю., Головченко А.А., Бочаров Д.М. Использование технологий искусственного интеллекта при игре в покер. Разработаны основные требования к работе покер-бота, представлены функции и характеристики системы искусственного интеллекта, разработан алгоритм поведения искусственного интеллекта.

Annotation:

Kotlubeyev A.Y., Golovchenko A.A., Bocharov D.M. Usage of artificial intelligence in poker games. The basic requirements for the work of a poker bot have been implemented, the functions and characteristics of the artificial intelligence system were presented, and an algorithm for the behavior of artificial intelligence has been developed.

Общая постановка проблемы

Искусственный интеллект (ИИ) в покере стал серьезной проблемой для всей индустрии карточных игр, а его отсутствие является конкурентным преимуществом, ведь начинающие игроки не способны победить искусственный интеллект. Подобные программы ежемесячно приносят убытки большим компаниям, измеряющиеся в миллионы долларов.

Таким образом, необходимо научиться распознавать примитивных покерных ботов, а также создать более развитые системы с использованием искусственного интеллекта.

Цель статьи

Определить основные требования и технологии для разработки системы искусственного интеллекта (бота) для игры в безлимитный техасский холдем покер.

Актуальность данной статьи

Разработки систем искусственного интеллекта крайне актуальны в современной жизни. Их развитие можно отследить на примере игр, которые не слишком сложны в разработке и не особо требовательны к ресурсам.

За последние десятилетия искусственный интеллект превзошел лучших мировых игроков в таких играх как: нарды, шашки, шахматы, Jeopardy! (англ. аналог «Своей игры»), в 2015 году – видеоигры Atari, в 2016 году – Го.

Однако, данные игры отличаются своей открытостью – игрок может анализировать состояние игры и положение соперника, в отличие от карточных игр, где состояние соперника остаётся неизвестным.

Никто не знает карт противника, блефует он или нет, какая карта выпадет на следующих этапах игры. Игрок может только предполагать это с определенной степенью вероятности.

Поэтому создание системы искусственного интеллекта для покера является актуальной задачей.

Общее описание игры

Состояние игры в покер можно разделить на:

- личную информацию игроков: две карты на руках, сданных в закрытую;
- публичную информацию: колода оставшихся карт, общий банк игры и банки игроков; последовательность ставок, сделанных игроками.

Возможные последовательности состояний в игре образуют дерево (рис.1), при этом каждое состояние имеет связанное поддерево.

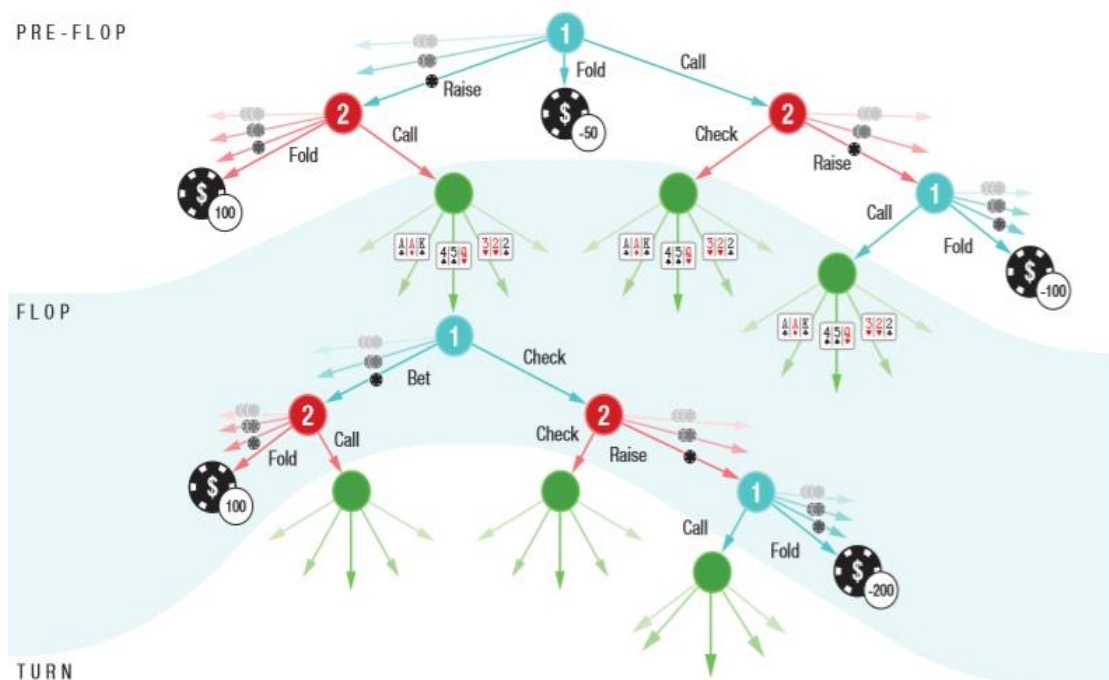


Рис. 1. Дерево последовательностей в покере.

Узлы представляют известную информацию, в то время как ребра представляют действия: две ветви показывают действия ставок игрока, а одна ветвь представляет общие карты, выпавшие случайно. Игра заканчивается на терминальных узлах со связанными значениями. Для терминальных узлов (где игрок не сбросил карты) игрок, чьи личные карты образуют более сильную руку, получает значение приоритетного игрока, т. е. решается для приближенного равновесия Нэша.

Стратегии игроков

Стратегия игрока определяется распределением вероятности допустимых действий для каждого шага принятия решения, где шаг принятия решения представляет собой комбинацию состояния игры и руки игрока. Учитывая стратегию игрока, для любого шага можно вычислить диапазон рук, который является распределенной вероятности по возможным картам игрока при достижении вскрытия.

Фиксируя стратегии обоих игроков, искусственный интеллект для конкретного игрока в состоянии вскрытия является билинейной функцией диапазонов обоих игроков с использованием матрицы выплат, определенной правилами игры. Ожидаемая полезность для игрока в любом другом публичном состоянии, включая начальное состояние, является ожидаемой полезностью по достижении вскрытия с учетом фиксированных стратегий игроков. Стратегия наилучшего ответа – это стратегия, которая максимизирует ожидаемую

полезность игрока против стратегии противника. В двухпользовательских играх с нулевой суммой, таких как HUNL, решение или стратегия равновесия Нэша максимизирует ожидаемую полезность при игре против оптимальной стратегии противника. Возможность использования стратегии – это разница в ожидаемой полезности по отношению к противнику с большей выгодой при равновесии Нэша.

Алгоритм вычисляет эту стратегию только для известной информации, которая возникает на каждом шаге. Несмотря на то, что стратегия вычисляется во время игры, она статична, хотя и стохастична, потому что это результат детерминированного вычисления, которое производит распределение вероятности по доступным действиям.

Компоненты алгоритма

Алгоритм состоит из следующих компонентов: расчет здоровой локальной стратегии для текущего шага, ограниченный по глубине видимой информации с использованием функции полученного значения, и ограниченный набор действий. На концептуальном уровне эти компоненты описывают эвристический поиск, который отвечает за успехи ИИ в играх с полной информацией. До алгоритма с неполной информацией не было известно теоретически обоснованного применения эвристического поиска в таких играх. Сердце эвристического метода поиска – это идея «непрерывного обратного поиска», где локальная процедура поиска вызывается всякий раз, когда игрок должен действовать, не сохраняя памяти, как или почему он ходил для достижения текущего результата. В основе алгоритма лежит непрерывное повторное решение, надежное локальное вычисление стратегии, которое требует лишь минимальной памяти о том, как и почему оно действовало, чтобы достичь текущего состояния.

Постоянное повторное решение

Предположим, были предприняты действия в соответствии с определенной стратегией, но затем при каком-то шаге данная стратегия была забыта. Возникает проблема реконструкции стратегии решения для поддерева без решения всей игры заново. Её можно разрешить через процесс повторного решения. Для этого нужно знать диапазон карт (включая общие) и вектор ожидаемых значений, достигнутых противником при предыдущем решении для каждой его руки. С этими значениями можно реконструировать стратегию лишь для оставшейся части игры, что не увеличивает общую нагрузку для принятия решения. Каждое значение в векторе противника является контрафактическим значением, т.е. условным значением «что-если», которое дает ожидаемое значение, если противник вскрывает карты с определенной рукой. Алгоритм также использует контрафактические значения и, если использовать его для принятия решения, то легко вычислить вектор противоположных контрафактических значений на любом шаге игры.

Повторное решение, однако, начинается со стратегии, в то время как цель состоит в том, чтобы никогда не поддерживать стратегию на протяжении всей игры. Данная проблема обходится путем постоянного реконструирования её путем переосмысления, когда нужно принять решения. Никогда не используется стратегия за пределами следующего действия. Для того чтобы повторно ее решить, нужно только отслеживать собственный диапазон рук и подходящий вектор противоположных контрафактических значений. Они должны быть пределом значений, которые противник может достичь каждой рукой в игре, будучи не больше, чем стоимость, которую мог бы отдать противник, чтобы их открыть. Это важная информация контрафактических значений, обычно используемых при повторном решении, с доказательством достаточности.

В начале игры, диапазон рук игрока одинаков с противником и контрафактуальные значения инициализируются к ручным картам. Когда наступает очередь хода, повторно решается поддерево в текущем положении игры, используя сохраненные значения диапазона

противника. Решение принимается в соответствии с вычисленной стратегией, отбрасывая прежнюю стратегию. После каждого действия либо игроком, либо общими картами, необходимо обновить диапазон и контрафактические значения противника. Обновляется и собственный процент выигрыша, используя вычисленную стратегию и правило Байеса. Следующий шаг: замена контрафактические значения противника на значения, вычисленные для этого шагом из последнего повторного решения. Обновляя собственный диапазон, удаляются карты, с которыми шанс победы минимален.

Достоинства и недостатки предложенного алгоритма

Описанные выше действия гарантируют, что противоположные контрафактические значения удовлетворяют достаточным условиям, и вся процедура производит произвольно близкие аппроксимации равновесия Нэша. Можно заметить, что постоянное повторное решение никогда не отслеживает диапазон противника, а только отслеживает их контрафактические значения. Кроме того, никогда не требуется знание решений противника для последующих шагов, что является важным отличием от традиционного повторения решений.

Постоянное повторное решение теоретически обосновано, но само по себе непрактично. Хотя оно никогда не поддерживает полную стратегию, повторное решение само по себе является трудноразрешимым, за исключением вскрытия. Для того, чтобы сделать постоянное повторное решение практическим, нам нужно ограничить глубину и ширину повторно решенного поддерева.

Выводы

Построение искусственного интеллекта для безлимитного тexasского холдем покера сложнее, нежели игры с полной информацией. При просчете принятия решения, кроме вероятности выпадение нужных общих карт, необходимо учитывать не только предположительную силу руки оппонентов, но и более сложные факторы для предугадывания закономерности принятий решений противника. Этими факторами могут быть: блеф, когда изображается сильная рука при слабой (либо же наоборот) и размер ставки, при которой продолжится либо завершится игра. Учитывая проблемы, упомянутые выше, в данной статье были определены основные требования и технологии для дальнейшей разработки системы искусственного интеллекта игры в безлимитный тexasский холдем покер, которая будет базироваться на равновесии Нэша. А также предусмотрена система постоянного повторного решения для более сложного предугадывания противником принятия решения ИИ.

Литература

10. Чен Билл. Математика покера / Б.Чен, Д.Анкенман. – Лондон: Conjel Co., 2012. – 232 p.
11. Дэвид Склански. Теория покера / Д.Склански. – Las Vegas: Creel Printing Co., 2005. – 638 p.
12. ИИ для покера: как научить алгоритмы блефовать [Электронный ресурс]. – Режим доступа: <https://habr.com/company/sberbank/blog/337264/>. – Загл. с экрана.
13. Matej Moravchik. DeepStack: Expert-Level Artificial Intelligence in Heads-Up No Limit Poker / M.Moravchik, M.Schmid, N.Burch. – Prague: Czech Technical University., 2017. – 37 p.

УДК 004.3

АЛГОРИТМ ПОСТРОЕНИЯ КАРТЫ ДЛЯ РОБОТА НА ОСНОВЕ ДАННЫХ УЛЬТРАЗВУКОВЫХ ДАТЧИКОВ

Егурнов Н.О., Миненко А.С.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: ygurnovn@gmail.com

Аннотация

Егурнов Н.О., Миненко А.С. Алгоритм построения карты для робота на основе данных ультразвуковых датчиков. В данной статье рассмотрен метод одновременной локализации и построения карты (SLAM), метод, используемый в мобильных автономных средствах для построения карты в неизвестном пространстве или для обновления карты в заранее известном пространстве с одновременным контролем текущего местоположения и пройденного пути.

Ключевые слова: SLAM, FastSLAM, DP-SLAM, мобильный робот, одометрия, фильтр Калмана, картография, локализация, навигация.

Abstract

Egurnov N.O., Minenko A.S. The algorithm for constructing a map for a robot based on data from an ultrasonic sensor. This article describes the method of simultaneous localization and mapping (SLAM), a method used in mobile offline tools for building a map in an unknown space or for updating a map in a previously known space while simultaneously monitoring your current location and distance traveled.

Key words: SLAM, FastSLAM, DP-SLAM, mobile robot, odometry, Kalman filter, mapping, localization, navigation.

Введение

Важной задачей в робототехнике является разработка способов определения положения робота в окружающем пространстве его навигации. Навигация в реалистичной среде является фундаментальным требованием для получения автономного мобильного робота. Не зная положения робота в пространстве невозможно решить задачу движения робота из одной точки в другую.

Реализация навигационной системы не представляет большой сложности при условии, что заранее известны ориентиры или карта пространства. Построение карты при постоянно известном положении робота также решаемая проблема, не вызывающая особой сложности. Гораздо сложнее столкнуться с заранее неизвестным положением робота и в неопознанном картой окружении. Целью данной статьи является рассмотрение теоретической основы. Ознакомление с основными методами и подходами к решению задачи SLAM. Автономная мобильная робототехника – интересная тема для исследования по многим причинам. Переход от компьютера на колёсах, который просто способен почувствовать некоторые физические свойства окружающей среды через её датчики в интеллектуальный агент, способный идентифицировать особенности, обнаруживать закономерности и учиться на опыте, локализовать, строить карты и ориентироваться.

Есть многие коммерческие предложения мобильных роботов. Транспорт, роботы для наблюдения, уборки или домашнего хозяйства. Автономные мобильные роботы, ещё на пути своего повсеместного, промышленного и бытового применения, в основном из-за отсутствия надёжных и гибких механизмов навигации и поведения. Мобильный робот должен

выполнять различные задачи, гарантирующие высокую степень надёжности. Установка маркеров, визуальных шаблонов облегчили бы эту проблему, но это дорого, не гибко и иногда просто невозможно. Продажа бытовых роботов с требованием установки искусственных маркеров во всём своём доме для управления не является чем-то практичным и тем более удобным. В большинстве случаев это невозможно, так как карта окружающей среды может быть не предоставлена вовсе. С другой стороны, оснащение робота точными и дорогими датчиками, для достижения высокой точности значительно увеличило цену робота, что сказалась бы на его широкомасштабном применении. Следовательно, робот должен быть в состоянии работать без какой-либо подготовки самой среды, используя минимальный набор датчиков, которые могут гарантировать надёжность и доступную цену. Представляет интерес исследовать методы, которые позволяют автоматически генерировать карты окружающей среды одновременно использовать их для надёжной локализации и навигации [1].

Представление структуры карты для методов локализации

В каком виде необходимо хранить карту это первая проблема, которую требуется решить, занимаясь задачей исследования местности. Что собой представляет карта, и в виде каких данных следует ее хранить. Прежде чем заниматься проблемой изучения местности нужно сформировать представление о том, как наилучшим образом описать окружающую среду, в которой будет функционировать автономный робот. Чаще всего местность можно условно разделить на две группы. Первая, с большим количеством ярко выраженных ориентиров. В качестве примеров среды первой категории можно привести поле с отдельно стоящими деревьями. Другим примером могут послужить здания с полостями строительных конструкций, будь то коридоры, комнаты или вентиляционная система, где в качестве ориентиров могут выступать углы. Вторая группа относит к себе среду, где ориентиров очень мало либо они трудноразличимы, или слишком велики и не могут быть использованы в качестве материальной точки. Если в исследуемой среде нет возможности найти ориентиры, то ее проще всего представлять в виде сетки с заполнением ячеек, занятых препятствиями. Хранить такую карту удобно в виде массива, где элементы, отражающие положение препятствий, имеют значение 1, а все остальные – 0. Такое представление карты применяется, например, в алгоритме DP-SLAM. В случае, когда в исследуемой местности есть многочисленные ориентиры, карта представляет собой массив оценок их местоположения. Для хранения структуры такой карты проще всего использовать картографическую базу данных, которая отражает положение ориентиров, их уникальные свойства и взаимосвязи.

Метод SLAM

Метод одновременной локализации и построения карты (SLAM) объединяет два процесса в последовательность алгоритмов и вычислений, при которых результаты одного процесса участвуют в вычислениях другого процесса. Метод, используется в мобильных автономных роботах для построения карты в неизвестном пространстве, или же для обновления карты в заранее известном месте. Одновременно контролирует текущее местоположения и пройденный путь. Популярные методы приближённого решения данной задачи включают в себя фильтр частиц и расширенный фильтр Калмана. Данные подходы используются в автопилоте автомобилей, беспилотных летательных аппаратов, автономных подводных аппаратов, планетоходов, домашних роботов. Обычно карты используются для определения позиции в пространстве и для графического изображения плана местности. Сложность технического процесса определения текущего местоположения и построения карты обусловлена низкой точностью приборов, участвующих в процессе вычисления текущего местоположения [2].

Фильтр частиц хорошо подходит для решения задачи локализации, где необходимо отслеживать позицию робота. Переход между состояниями – это движение робота, а наблюдения – показания дальномера. Обе эти величины имеют большое количество различных помех и неточностей для этого и применяются алгоритмы фильтрации.

Для определения движения в качестве базиса моделирования выбирается перемещение, измеренное системой одометрии. С использованием одометрии возможно очень точно измерять угол поворота колёс, однако определить реальные перемещения робота с её помощью проблематично, так как любые погрешности, связанные с проскальзыванием, сдвигами и неровностями поверхности, по которой перемещается робот могут стать причиной погрешностей в вычислениях и дальнейших ошибок, накопление приведёт к полному несоответствию данных действительности.

Модели движения для различных типов поверхностей серьёзно отличаться. Для этого и используются дальномеры, определяющие расстояния к ближайшим препятствиям, на основе которых строятся частицы. Посредством моделирования карты необходимо взвесить все частицы согласно текущим наблюдениям робота. При решении проблемы чистой локализации робот имеет полную карту в памяти, но чаще всего она является неполной в той или иной степени. Позиция, описанная каждой частицей, соответствует некоторой известной точке, препятствию зарегистрированной дальномером или иным другим способом отмечая определенное направление на карте. Исходя из этого, возможно легко выявить, какие показания должен вернуть дальномер, если робот находится в данной позиции.

Классические подходы к решению данной задачи с использованием фильтра частиц предполагают составление единой карты для всех позиций робота. Такие подходы обладают существенным недостатком – ошибки каждой частицы привносят вклад в общую карту и накапливаются там с течением времени. Из этого можно сделать вывод, что все наблюдения должны сравниваться не с общей картой, а с неполной возможно некорректной картой, хранящейся в каждой частице. Итоговой же картой является карта частицы с наибольшим весом, остальные отбрасываются.

Такой подход является частным решением поставленной задачи. Однако на практике частицы с ошибками в оценке положения робота будут добавлять ошибочные данные в свою карту. Ошибка на карте, в свою очередь, породит ошибку локализации на следующем шаге, и эти неточности будут накапливаться очень быстро. Возникает так же проблема необходимости хранения и обработки большого множества частиц, сохраняя высокую производительность. Учитывая, что каждая частица – это не просто позиция робота, но ещё и карта исходя из этого хранение сотен, а то и тысяч карт представляет значительные трудности. Одним из подходов к решению этой проблемы является допущение, что неопределённость на карте может быть представлена в простой параметрической форме.

Метод FastSLAM

Этот подход, применяется в алгоритме FastSLAM, для которого карта представляет собой фильтр Калмана над множеством меток, расставленных в известных точках пространства.

Существует так же наивный подход, в котором каждая частица соответствует некоторой траектории в пространстве и имеет свою собственную карту. Когда одна частица порождает другую, вся карта целиком рассматривается как часть скрытого состояния и копируется в новую частицу. В то же время для большого количества частиц, достаточного для точной локализации и крупной карты разумного размера, этот подход потребовал бы перемещений гигабайтов данных на каждой итерации, что не практично.

Сложностью реализации карты является тот факт, что необходимо разработать структуру данных, которая бы работала одинаково быстро как с обновлениями, так и с запросами локализации. Наивный подход, в котором каждая частица имеет свою

собственную карту, неэффективен. Чуть менее наивное решение, в котором сохраняются только изменения карты также неэффективно, так как зависит от количества итераций фильтра.

Метод DP-SLAM

Альтернативной является алгоритм DP-SLAM, в котором реализован альтернативный подход. В нем карты и частицы меняются местами. DP-mapping хранит только одну карту. В отличие от классических подходов, каждая ячейка карты содержит сбалансированное дерево. Каждый узел такого дерева проиндексирован идентификатором частицы, определяя тем самым наличие препятствий на пути или их отсутствие.

Сетка карты проинициализирована матрицей пустых деревьев. Когда частица намеревается добавить наблюдение о ячейке сетки, она добавляет свой идентификатор в соответствующее дерево. Преимущество этого метода в том, что он позволяет каждой частице вести себя так, как будто она имеет свою собственную карту. Чтобы проверить значение ячейки, частица проверяет каждого предка и ищет последнего изменившего данную ячейку. Если дерево, содержащееся в ячейке, пусто, то частица считает её состояние неизвестным.

Теперь мы можем описать объединение предка с единственным потомком более подробно: вначале множество измененных потомком ячеек объединяется с множеством его родителя. Затем для каждой ячейки, измененной потомком, хранящийся в сбалансированном дереве идентификатор меняется родительским. После этого частица-потомок удаляется из дерева наследования, и родительские внуки становятся его прямыми наследниками. Стоит заметить, что это обеспечивает ограниченность сверху количества элементов дерева каждой ячейки, что в конечном итоге освобождает память от лишних ненужных данных и использует только то, что требуется на данный момент [3].

Выводы

В данной статье были рассмотрены особенности и функции SLAM. Проведен обзор существующих алгоритмов SLAM в двухмерном пространстве. Рассмотрены основные преимущества и недостатки различных методов, подходы к решению задачи SLAM для мобильного робота. Был предложен алгоритм решения задачи одновременной локализации и построения карты для мобильного робота на основе данных, получаемых с ультразвуковых дальномеров.

Литература

1. Guido Zunino "Autonomous Domestic Navigation. Simultaneous Localization and Mapping for Navigation in Realistic Environments" Royal Institute of Technology Numerical Analysis and Computer Science [Текст] / G. Zunino // KTH, Stockholm 2002
2. Википедия SLAM (метод) [Электронный ресурс] / Интернет-ресурс. - Режим доступа: <https://ru.wikipedia.org/wiki/SLAM>
3. Duke Computer Science, DP-SLAM Austin Eliazar and Ronald Parr [Электронный ресурс] / Интернет-ресурс. - Режим доступа: <https://users.cs.duke.edu/~parr/dpslam/>

УДК 004.89

РАЗРАБОТКА ОНТОЛОГИЧЕСКОГО ЧАТ-БОТА НА БАЗЕ ГЛУБИННОГО ОБУЧЕНИЯ НЕЙРОННЫХ СЕТЕЙ

Оверченко Я.Ю., Копытова О.М.

Донецкий национальный технический университет

E-mail: yarikoverchenko@gmail.com, omkop@list.ru

Аннотация

Оверченко Я.Ю., Копытова О.М. Разработка онтологического чат-бота на базе глубинного обучения нейронных сетей. Рассмотрены основные этапы процесса разработки онтологического чат-бота, основанного на нейронной сети. Приведено обоснование выбора архитектуры и метода обучения нейронной сети. Описаны некоторые особенности процедуры обучения чат-бота, построенного на основе нейронной сети.

Ключевые слова: чат-бот, нейронные сети, глубинное обучение.

Abstract

Overchenko Y., Kopytova O. Development of ontological chatbot based on deep learning of neural networks. The main stages of the development of an ontological chatbot based on a neural network are considered. The rationale for the choice of architecture and method of learning of the neural network. Described are some features of the chat bot training procedure built on the basis of a neural network.

Key words: chat bot, neural networks, deep learning.

Общая постановка проблемы

Чат-боты – это компьютерные системы, которые позволяют автоматизировать различные работы, связанные с использованием сети Интернет, такие как, например, регистрация заявок пользователей или обеспечение предварительной технической поддержки, проведение опросов пользователей определенной сети. Помимо того, чат-боты могут выступать помощниками в обучении, автоматизируя процесс подачи различной информации. Эти компьютерные системы в зависимости от сферы применения могут быть основаны на заранее подготовленных шаблонах или иметь более сложную архитектуру и базироваться на использовании нейронных сетей с применением технологий искусственного интеллекта.

Процесс разработки чат-бота, основанного на нейронной сети, включает в себя несколько этапов, таких как выбор архитектуры и реализация нейронной сети, подготовка данных для её предварительного обучения и непосредственно само обучение [4]. Далее будет приведено краткое описание этапов разработки нейросетевого чат-бота и их особенностей.

Выбор архитектуры и реализация нейронной сети

На данном этапе существует два варианта его выполнения – собственная реализация нейронной сети и использование готовой подходящей реализации одного из специализированных фреймворков с её последующей настройкой для разрабатываемой системы.

В ходе исследования вариантов собственной реализации нейронной сети была проверена эффективность трех различных архитектур, схема которых приведена на рисунке 1.

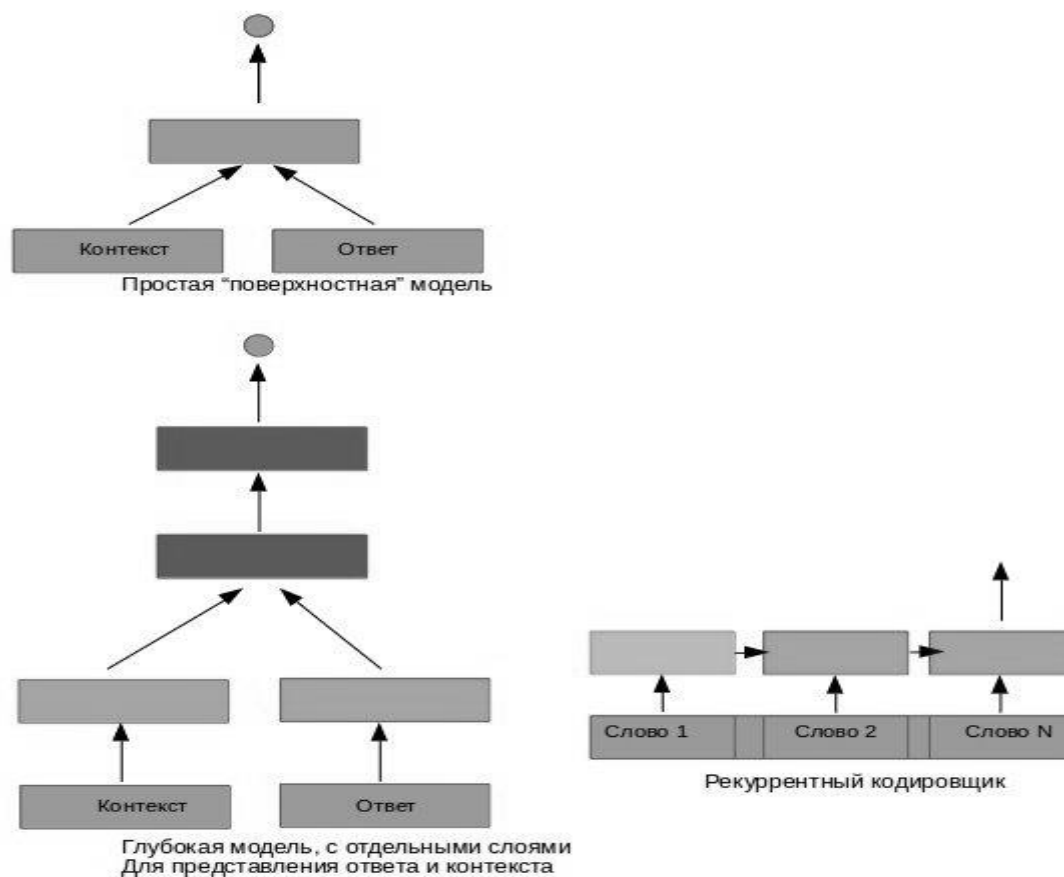


Рисунок 1 – Схемы архитектур нейронных сетей

Первая модель (простая «поверхностная») берет соединение последовательности векторов слов контекста и ответа и подает все это на обычный полностью соединенный слой. Поскольку ответ бывает разной длины, его записываем в вектор длины фиксированной, заполняя «лишнее» место нулями [5].

Вторая глубокая модель отличается тем, что до «смешивания» контекста и ответа им дан «свой собственный» слой для формирования представления.

Третья модель кодирует последовательность разной длины с помощью рекуррентного LSTM кодировщика. Это много медленнее и обучается дольше. Но теоретически данная архитектура должна показать наилучший результат [5].

В ходе проведения тестирования архитектур по способности выбирать правильный ответ из нескольких предложенных вариантов (случайно взятых ответов), самый лучший результат показала модель номер 2, с пятью слоями для анализа данных по 700 нейронов в каждом (+1 слой репрезентации входных данных, тоже из 700 нейронов). Чтобы обучить такую глубокую сеть, пришлось использовать Хайвей (Highway) слои, которые позволяют информации проще распространяться от нижних слоев к верхним. Модель номер 1 явно недообучилась, а номер 3 долго обучалась, и тоже не добралась до результатов модели номер 2 несколько процентов.

Также на данном этапе существует вариант использования готовой реализации нейронной сети с её дальнейшей настройкой для своих целей. Данный подход может быть использован в случае, если для выполняемой задачи уже существует подходящая реализация нейронной сети и создавать такую же самостоятельно не имеет смысла, если она не будет иметь каких-либо отличительных особенностей или прироста в производительности.

Наиболее распространенным фреймворком нейронных сетей на данный момент является TensorFlow. Для реализации чат-бота, данная платформа содержит модель RNN (рекуррентной нейронной сети), которая также используется для обучения модели перевода для пары языков английский / французский [4].

Подготовка данных для обучения

Важность данного этапа заключается в том, что от качества данных для обучения зависит качество и уровень осмысленности ответных реплик, которые будет генерировать разрабатываемый чат-бот.

Стандартно при реализации подобных систем разработчики используют базы субтитров к фильмам. Такая база есть и для русского языка, правда меньше размером. Но большая проблема подобных баз заключается в том, что в них много монологов, излишней информации или слабая смысловая связь реплик, а также сложно разделить диалоги друг от друга. В ходе формирования обучающих данных было решено дополнить их собственными диалогами из социальной сети «ВКонтакте». С помощью специального клиента их можно скачать в подходящем формате вопрос-ответ.

Обучение нейронной сети

Данный этап выполняется, в основном, автоматически, однако требует большой вычислительной мощности аппаратных средств. Обучение можно производить, выполняя вычисления с использованием центрального процессора компьютера или его видеокарты.

В случае если домашний компьютер не справляется с задачей обучения, необходимо использовать специальные серверы, которые обладают мощным оборудованием и способны выдержать длительную высокую нагрузку на аппаратные средства. Длительность процесса обучения нейронной сети зависит от количества слоёв в её архитектуре и объёма обучающих данных.

Выводы

Были рассмотрены основные этапы процесса разработки чат-бота, основанного на базе нейронной сети. Также были описаны и проанализированы три наиболее подходящие архитектуры нейронной сети, в ходе тестирования которых была выявлена наиболее эффективная модель. Были рассмотрены особенности и варианты создания обучающих данных, а также процесса обучения с их помощью.

Литература

14. Савельев, А. В. На пути к общей теории нейросетей. К вопросу о сложности // Нейрокомпьютеры: разработка, применение / А.В. Савельев. – 2006. – № 4–5. – С. 4–14.
15. Николенко, С. Глубокое обучение / С. Николенко, А. Кадурин, Е. Архангельская. – СПб.: Питер, 2018. – 480 с.
16. Черняк, Е. Глубинное обучение в обработке и анализе текстов [Электронный ресурс]: авт. к.т.н. – <https://postnauka.ru/longreads/85951>.
17. Ковалевский В.. Chatbot на базе рекуррентной нейронной сети своими руками [Электронный ресурс]: – <https://m.habr.com/ru/post/317732/>
18. Тарасов Д.. Русский нейросетевой чатбот [Электронный ресурс]: – <https://m.habr.com/ru/company/meanotek/blog/280268/>

УДК 519.683.8

РАЗРАБОТКА ОБУЧАЮЩЕГО ПАКЕТА ПРОГРАММ ПО КУРСУ «ОПЕРАЦИОННЫЕ СИСТЕМЫ»

Ищенко Р.Д., Ольшевский А.И

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: darkshonjonson@gmail.com, a_olshevskiy@mail.ru

Аннотация:

Ищенко Р.Д., Ольшевский А.И. Разработка обучающего пакета программ по курсу «Операционные системы». Рассмотрены основные положения курса «Операционные системы». Рассмотрены задачи, которые необходимы для разработки обучающего пакета программ по данному курсу. И краткое описание о реализации программных пакетов

Annotation:

Ishchenko R. D., Olshevsky A. I. Development of a training program package for the course "Operating systems". The main provisions of the course "Operating Systems". The tasks that are necessary for the development of a training package for this course are considered. And a brief description of the implementation of software packages.

Общая постановка проблемы

Передовые страны проводят новую образовательную политику, опирающуюся на инженерный подход, связанный с индивидуализацией процесса обучения. Реализация такой обучающей системы возможна на основе результатов следующих исследований в области искусственного интеллекта.

Для высшего образования первостепенную актуальность приобретает задача использования возможностей информационных технологий в обучении. На сегодняшний день мало существует пакетных продуктов для индивидуального изучения специализированных курсов. Студентам, находящимся на заочной форме обучения, сложно углубленно изучать новые IT-технологии самостоятельно.

Цель статьи

Накопление и систематизация информации по курсу «Операционные системы», формирование у пользователей аналитико-мыслительной деятельности, повышение эффективности процесса обучения.

Актуальность данной статьи

Курс «Операционные системы» относится к базовой части дисциплин профессионального цикла подготовки студента.

Разработка данной системы будет актуальна в случае самостоятельного изучения предмета студентом.

Описание концепции обучающей системы поможет понять технологии создания и организации современных операционных систем.

Для обучения будут применены мультимедийные технологии, которые обеспечивают лучшее восприятие учебного материала. Иллюстрации являются зрительной опорой, которая помогает наиболее полно усвоить материал.

Вводная информация о курсе

Операционные системы – это программное обеспечение для администрирования аппаратного и программными ресурсами компьютера и предоставляющее общие службы для программ. Также операционные системы предназначены для взаимодействия с пользователем.

Самые распространенные операционные системы для персональных компьютеров, это в первую очередь семейство Windows от корпорации Microsoft и семейство unix подобных систем Linux.

Сама операционная система – это основная часть программного обеспечения, работающая в режиме супервизора. Данный режим работает с аппаратным обеспечением и имеет к нему полный доступ и может задействовать любую инструкцию. Другие программные продукты работают в режиме пользователя и имеют ограниченный доступ к аппаратной части. Например, не могут использовать инструкции ввода – вывода.

Программы пользовательского интерфейса – это оболочка для операционной системы, в которой пользователь работает с пользовательскими программами, что означает, что пользовательский интерфейс (графическая оболочка или GUI) находится на низком уровне программного обеспечения.

Операционные системы возникли с сороковых годов двадцатого века. С того времени по наши дни было разработано множество операционных систем. Опишем некоторые из них:

- мейнфреймовые;
- серверные;
- многопроцессорные;
- операционные системы персональных компьютеров;
- операционные системы карманных персональных компьютеров;
- встроенные операционные системы;
- операционные системы сенсорных узлов;
- операционные системы реального времени;
- операционные системы смарт-карта.

Большая часть операционных систем используют такие понятия как процессы, адресные пространства и файлы для понимания работы и возможности объяснить их принцип работы.

Процессом в операционных системах является программа во время её выполнения. С каждым процессом связано его адресное пространство место в памяти компьютера. Он является одной из самых важных абстракций в операционной системе. Процессы помогают осуществить псевдо – параллельные операции. Без абстракции процессов современные вычисления просто не могут существовать.

Адресным пространством в операционных системах является совокупность всех допустимых ячеек памяти. Данные ячейки могут быть использованы для доступа к этим объектам при определенном режиме работы.

Файлы – это логический информационный блок, создаваемый процессами. Для удобства работы с информацией, а в частности с файлами, начали появляться структуры, которые определяли способ организации, хранения и именования данных на носителях информации.

Решаемые задачи

В общем случае в рамках автоматизированных обучающих систем могут решаться следующие задачи:

- задачи, связанные с регистрацией и статистическим анализом показателей усвоения учебного материала: определение времени решения задач, определение общего числа ошибок и т.д.; к этой же группе относятся и задачи управления учебной деятельностью;
- задачи, связанные с проверкой уровня знаний, умений и навыков учащихся до и после обучения, их индивидуальных способностей и мотиваций;
- задачи АОС, связанные с подготовкой и предъявлением учебного материала, адаптацией материала по уровням сложности, подготовкой динамических иллюстраций, контрольных заданий, лабораторных работ, самостоятельных работ учащихся;
- задачи администрирования системы, доставки учебного материала на рабочие станции и задачи обратной связи с обучаемым.

Для создания обучающего модуля в общем случае, решаются следующие задачи:

- подготовки и систематизации учебного материала, адаптации материала по уровням сложности, разработки динамических иллюстраций, контрольных вопросов и других заданий (база знаний);
- регистрации пользователей (создания модели обучаемого), проверки уровня знаний и статистического сбора показателей усвоения учебного материала;
- оценки уровня знаний, умений и навыков у обучаемых до и после обучения, их индивидуальных способностей и мотиваций;
- администрирования системы, доставки учебного материала на рабочие станции и задачи обратной связи с обучаемым.

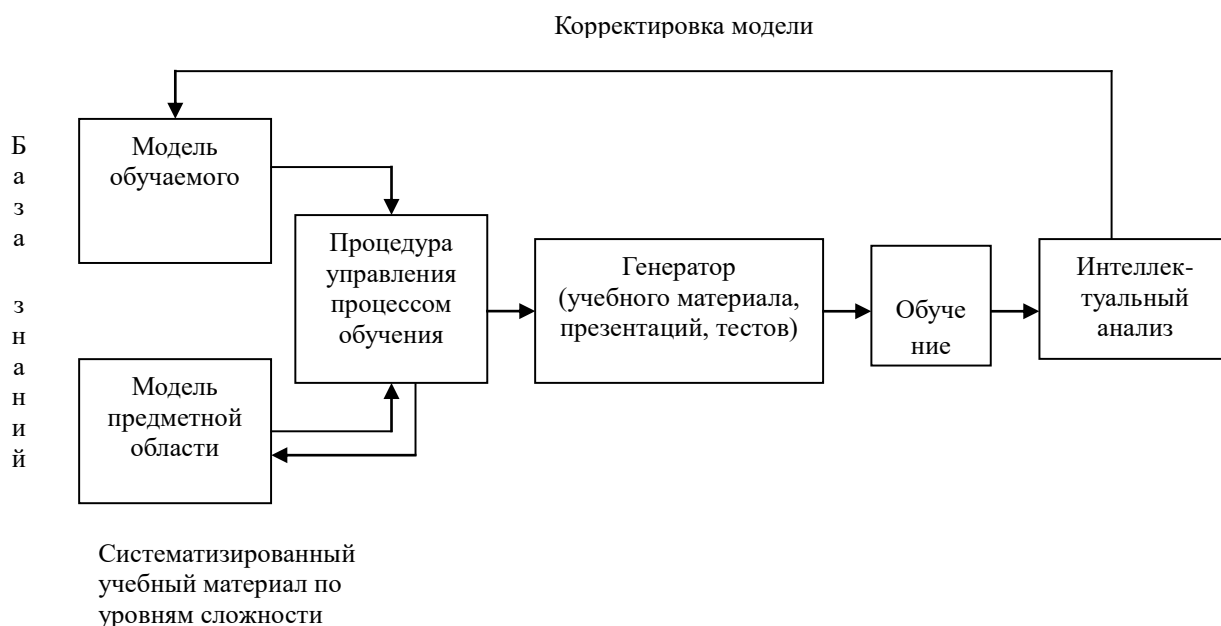


Рисунок 1 – Корректировка модели

Интеллектуальный анализ результатов – это процедура проверки, при которой обучающая система способна проанализировать ответы студента на тестовые задания, указать, что именно неправильно или неполно освещено в ответе и, как следствие, какие знания недостаточно усвоены студентом.

Центральное место в системе занимает процедура управления процессом обучения, которая используется для планирования и выбора дальнейших действий (логические «решатели», методы теории принятия решений). Для генерации новой порции учебного

материала и помощи обучаемому задействуется база знания о предметной области и текущее состояние модели обучаемого. Сведения извлекаются по запросу анализатором действий системы управления. После удачного прохождения всех этапов обучаемому предлагается следующая тема из этого курса.

Метазнания о компетенции обучаемого первоначально формируется путём диагностирования обучаемого с помощью тестов, а также на умение выполнять простейшие или более сложные упражнения. При этом осуществляется последующее итеративное уточнение компетентности модели обучаемого по результатам его ответов в ходе изучения им очередной порции материала, предложенной ему с учётом достигнутой компетенции.

Описание программы.

Данные программные пакеты будут реализованы при помощи языка Python и фреймворка Flask, так как была выбрана архитектура клиент – сервер. Также будет реализована база данных для хранения информации о пользователе и его прогрессе в изучении данного курса. Для этого будет реализована авторизация пользователей, при помощи логина и пароля.

В базу данных будут входить такие таблицы как, пользователь, тесты, учебные материалы.

В таблице пользователь будут находиться такие поля как, логин, пароль, пройденные темы, оценка за тему и средняя оценка.

В таблице учебные материалы будут находиться ссылки на иллюстрации, видеоматериалы и текстовые описание.

В таблице тесты будут находиться данные для тестирования пользователя и их оценивания.

Программный интерфейс будет реализован при помощи HTML и шаблонизатора jinja2.

Выводы

В данной статье была систематизирована информации по курсу «Операционные системы», описана вводная информация, определены решаемые задачи и описана корректировка модели обучения. Также было дано краткое описание структуры программы.

Литература

1. Таненбаум, Э. Современные операционные системы / Э.Таненбаум, Х.Бос. – 4-е изд. – СПб.: Питер, 2015. – 1120 с.
2. Руссинович, М. Внутреннее устройство Microsoft Windows / М. Руссинович, Д. Соломон. – 6-е изд. – СПб.: Питер, 2013. – 800 с.
3. Кофлер, М. Linux. Полное руководство / М. Кофлер. – СПб.: Питер, 2011. – 800 с.
4. Голощапов, А. Л. Google Android: системные компоненты и сетевые коммуникации / А.Л Голощапов. – СПб.: БХВ-Петербург, 2012. – 384 с.
5. Олифер, Н. А. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов / Н. А.Олифер, В. Г. Олифер. – 4-е изд. – СПб.: Питер, 2011. – 944 с.
6. Олифер, Н. А. Сетевые операционные системы / Н. А.Олифер, В. Г. Олифер. – СПб.: Питер, 2007. – 864 с.

УДК 004.031.42

ОПРЕДЕЛЕНИЕ ТРЕБОВАНИЙ К РАЗРАБАТЫВАЕМОМУ ИНТЕРНЕТ-МАГАЗИНУ

Проничев А.А., Безрукавый И.В., Семенова А.П., Бочаров Д.М.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: pronichevtolik@gmail.com

Аннотация

Проничев А.А., Безрукавый И.В., Семенова А.П., Бочаров Д.М. Определение требований к разрабатываемому интернет-магазину. Данная статья посвящена определению требований к функциональности, интерфейсу, составу интернет-магазина и выделению основных системных требований к хостингу.

Abstract

Pronichev A.A., Bezrukavyy I.V., Semenova A.P., Bocharov D.M. Determination of requirements for the developed online store. This article is devoted to the definition of requirements for functionality, interface, composition of the online store and the allocation of basic system requirements for hosting.

Общая постановка проблемы

При разработке любого интернет-сайта, а особенно, интернет-магазина в первую очередь необходимо определить набор требований к функциональности, интерфейсу, составу программного обеспечения, поскольку оно является лицом предприятия и должно привлекать, а не отталкивать потенциального пользователя.

Кроме того, определение требований может сильно повлиять на выбор хостинга, на котором должен размещаться разрабатываемый интернет-сайт.

Функциональные требования

Интернет-магазин должен позволять выполнять следующие функции:

- инициализацию системы (ввод перечней товаров в соответствии с торговыми планами, а также списков покупателей и т. п.);
- ввод и коррекцию текущих данных о выполнении и обработке заказов;
- хранение данных о покупателях;
- получение информации о текущем состоянии товаров на складе.

Исходные данные:

- наличие товара на складе;
- выбранная покупателем модель изделия;
- текущая информация о возможности доставки товара.

Результаты:

- запись информации о покупателе в базу данных (БД);
- оформление договора о покупке товара;
- оплата товара;
- заказ товара у поставщиков в случае его отсутствия на складе;
- доставка товара покупателю.

Требования к интерфейсу

С обширным распространением интернета огромную популярность получили веб-интерфейсы, которые позволяют взаимодействовать с различными программами через веб-браузер (например, управление заказами в интернет-магазине). Веб-интерфейсы очень удобны еще и тем, что предоставляют возможность вести совместную разработку сотрудникам, находящимся в разных офисах (например, веб-интерфейсы часто применяют для заполнения баз данных, а также публикации материалов в сети интернет).

В широком смысле интерфейс представляет собой определенную стандартами границу между независимыми объектами, обменивающимися между собой некоторой информацией. Интерфейс задает параметры объектов, а также их процедуры и характеристики взаимодействия. Интерфейс пользователя – это компоненты и элементы программы, способные оказывать непосредственное влияние на взаимодействие пользователя системы с самим программным обеспечением. Он включает в себя:

- командные режимы, язык пользователь-интерфейс;
- средства отображения информации, сама отображаемая информация;
- различные форматы и коды;
- диалоги, транзакции и взаимодействие между пользователем и системой;
- технологии и устройства ввода данных;
- обратная связь с пользователем системы;
- документация, порядок использования программы;
- поддержка принятия решений в определенной предметной области.

Существует несколько простых правил, которые позволяют интерфейсу интернет-магазина быть простым и понятным клиенту.

1. Чем проще, тем лучше. Это не означает, что на сайте должен быть только текст и контакты, но на нем не должно быть ненужной информации, к тому же шрифт должен легко читаться. Используемые графические элементы должны быть выразительными, четкими и быстро загружаться. На сайте интернет-магазина нельзя применять звук и анимацию, которые долго загружаются, отвлекая внимание потенциального покупателя. Человеческий глаз пролистывает страницы интернет сайта сверху вниз. Самое большое внимание при этом сосредотачивается на его верхней левой части. Следовательно, в верхней части страницы обычно размещают наиболее важную информацию, такую, как: логотип, название фирмы, название сайта и т.п. Чем проще сделана верхняя часть страницы магазина, тем легче пользователю запомнить название сайта и, собственно, саму фирму. Типичной ошибкой множества сайтов является нагромождение вверху страницы сайта сложной композиции из рисунков, текста, среди которых может быть едва заметно само название фирмы и сайта. При этом человеческий глаз при сканировании данной части страницы интернет-магазина, не успеет распознать и запомнить важную информацию. Таким образом, происходит перенасыщение восприятия. Вместо увеличения запоминаемости происходит снижение запоминаемости основных надписей сайта, таких, как название фирмы и сайта или другой важной информации.

2. Потенциальный покупатель должен легко находить интересующий его товар, а также иметь возможность получать о нём полную информацию (фотографии, видео обзоры, текстовые обзоры).

3. Товары в интернет-магазине должны быть распределены по логически структурированным группам. Необходимо обеспечить возможность быстрой и понятной навигации по товарным группам и товарам.

4. Оформляя заказ, покупатель обычно вводит такую контактную информацию: имя, e-mail, город, страна, телефон и т.д. Не нужно сильно перегружать пользователя требованиями ввести множество полей, поэтому приведенных выше данных достаточно.

Остальные данные уточнит менеджер после того, как перезвонит клиенту для подтверждения заказа.

5. Расчёт стоимости товаров и вывод цен должен осуществляться в национальной валюте с возможным применением других валют при необходимости. Чтобы сэкономить время клиента на оплату товара, можно скомпоновать готовую накладную для оплаты через банк и отправить ее клиенту на электронную почту после оформления заказа.

6. В интернет-магазине могут быть такие информационные разделы:

- с информацией о доставке товара;
- с информацией о самом магазине (сфера деятельности, контактные телефоны, адрес и т.д.);
- с информацией об оплате товара;
- SEO статьи для продвижения через поисковые системы, а также для информирования пользователей о новинках и другой важной информации.
- иная полезная информация.

7. «Вопросы и отзывы» по товарам. Покупателю нужно предоставить возможность просмотреть все отзывы об интересующем товаре, а также оставить свой отзыв либо задать вопрос. Все это способствует повышению качества сервиса.

8. Росту посещаемости интернет-магазина и числа постоянных клиентов способствует также использование обратной связи администратора с пользователями магазина. Важно, чтобы администраторы сайта как можно быстрее и максимально корректно отвечали на замечания и отзывы клиентов.

9. Невидимой, обратной покупателю, стороной онлайн магазина является его система управления. Вход в админ панель осуществляется после ввода работником-администратором запрашиваемого системой логина и пароля. Администратор сайта должен иметь возможность полностью управлять контентом интернет-магазина, а именно:

- редактировать все разделы магазина (статьи, вопросы и ответы, отзывы и вопросы к товарам, новости и т.д.);
- добавлять или удалять товары, а также описание и фотографии к ним;
- менять стоимость, условия оплаты и доставки, уровень скидок;
- редактировать контактную информацию;
- редактировать содержание текстов и заголовков электронных писем, которые отправляются покупателю при заказе товара;
- просматривать статистику покупателей и историю заказов;
- изменять курсы валют на витрине интернет-магазина.

10. Продуманная работа с цветовой гаммой. Грамотно примененный цвет может, к примеру, передавать несущественные различия между однородными элементами веб-интерфейса. В то же время, неправильно использованный цвет может наоборот мешать работать с сайтом. Обычно пользователь просто закрывает такой сайт и не возвращается к нему в будущем. Поэтому цветовая гамма – это наиболее важный параметр в разработке веб-интерфейса сайта. Особое внимание следует уделять красному цвету. Для большинства людей красный цвет ассоциируется с опасностью. При чрезмерном использовании красного цвета в каком-либо элементе веб-интерфейса, эта область на экране привлекает слишком много внимания и часто заставляет пользователя настораживаться. Следовательно, если кнопка на экране выкрашена красным, пользователь возможно будет стараться избегать кликать на нее. В то же время, в незначительных количествах красный цвет может служить в качестве указания наличия каких-либо проблем. К примеру, если введенная пользователем информация не соответствует какому-либо шаблону. Красный цвет также может применяться в комбинации с другими цветами.

Существуют две метафоры: «термометр» и «светофор». «Термометр» означает, что красному цвету противопоставлен синий, а при использовании «светофора» красному

противопоставлен зеленый. Обе эти метафоры должны быть использованы только, если это необходимо. Руководствуясь приведенными выше принципами разработки веб-интерфейса, принято решение сделать ставку на информативность и простоту, чтобы покупатель, попадая на сайт интернет-магазина, получал четкую информацию о продукте, а также о способах его оплаты и доставки. В электронном магазине также должна быть реализована удобная и быстрая навигация для доступа пользователя к требуемому товару, так как не все пользователи имеют неограниченный доступ в сеть интернет, особенно используя мобильные телефоны и планшеты. Таким образом, стиль сайта должен быть:

- простой (минималистический), легкий, удобный;
- впечатление, которое должен произвести веб-сайт на пользователя;
- качественная продукция, понятная и простая навигация;
- в результате посещения магазина пользователь должен;
- увидеть необходимые запчасти, полезную информацию, почувствовать удобство, уверенность в профессионализме продавца;
- наличие фирменного стиля/бренда компании;
- специальный фирменный стиль отсутствует.

Должна быть предусмотрена система заказа товаров, отсутствующих на складе магазина. Цветовая гамма пользовательского интерфейса должна быть ориентирована на светлые серо-розовые тона, которые удачно гармонируют с темным цветом текста.

Требования к составу страниц

Приведем обязательные элементы для всех страниц:

- шапка сайта;
- ссылка на корзину;
- ссылки на разделы с информацией о способах оплаты, доставки, а также контакты и информацию о магазине.

Структура разделов сайта:

- 1) О нас;
- 2) Каталог товаров;
- 3) Корзина;
- 4) Личный кабинет;
- 5) Оплата и доставка;
- 6) Контактные данные;
- 7) Прайс лист товаров.

На главной странице должно быть:

- меню;
- каталог товаров;
- информация;
- лидеры продаж

Выводы

Исходя из существующих требований к разрабатываемому интернет-магазину можно выделить основные системные требования. Системные требования к интернет-магазину – это набор характеристик компьютера, которые обеспечивают работу определенного программного обеспечения. Чаще всего это следующие сведения:

- операционные системы, на которых может работать ИС (в подавляющем большинстве на текущий момент это версии Windows);
- процессор (модель, тактовая частота, производитель, или рейтинговое число для процессоров AMD);
- оперативная память (RAM), например, 1000 Мб (1 Гб);

- требования к видеокарте (производитель графического процессора его– модель, объем видеопамяти);
- необходимое пространство на жестком диске, например, 5 Гб свободного пространства;
- иногда указываются требования к звуковым картам;
- могут быть требования к составу модулей и сервисов ОС.

Системные требования бывают минимальные и рекомендованные. Таким образом, для нормальной работы с ИС рабочие места пользователей должен иметь характеристики не ниже указанных:

- операционная система Windows XP/VISTA/7;
- установленный браузер, желательно Google Chrome не ниже 10-й версии, IE не ниже 10-й версии;
- ОЗУ 512 Мб и выше;
- процессор частотой не ниже 1 ГГц;
- видеокарта с видеопамтью не менее 128 МБ;
- сетевая карта.

Поскольку разрабатывается интернет сайт, то немаловажными являются также требования к хостингу, на котором данный сайт будет размещаться. К этим требованиям относится:

- Apache / Windows IIS / NGINX;
- PHP 5.2 и выше;
- MySQL 5;
- mCrypt;
- php-xml (для работы vQmod);
- php-xmlrpc (для работы vQmod);
- cron расписания.

Литература

1. Конверс, Т. PHP5 и MySQL. Библия пользователя / Т. Конверс, Д. Парк, К.Морган. – М. : Диалектика, 2006. – 1216 с.
2. Балабанов, И.Т. Электронная коммерция / Балабанов И.Т. – СПб : Питер, 2001 – 336с.
3. Юрасов, А.В. Электронная коммерция: учебное пособие / Юрасов А.В. – М. : Дело, 2003. – 480 с.
4. Голоскубова, Н. Интернет-магазин: організація, опадоткування, облік / Голоскубова Н., Клиженко Я., Пустовіт М. – Х. : Фактор, 2009. – 128с.
5. Лещев, Д. Создание интерактивного web-сайта: учебный курс / Лещев Д. – СПб. : Питер, 2003. – 544 с.
6. Печников, В.Н. Создание Web-страниц и Web-сайтов. Самоучитель / Печников В.Н. – М. : Изд-во Триумф, 2006. – 464 с.
7. Хокинс, С. Администрирование Web-сервера Apache и руководство по электронной коммерции / Хокинс С. – М. : Вильямс, 2001. – 336 с.
8. Ульман, Л. Основы программирования на PHP / Ульман Л. – М. : Пресс, 2001. – 288с.
9. Томас, М. Секреты программирования для Internet / Томас М., Пател П. – СПб. : Питер Пресс, 1997. – 640с.
10. Будилов, В. А. Основы программирования для Интернета / Будилов В. А. – СПб.: БХВ-Петербург, 2003. – 736 с.

11. Холыднер, С. PHP в примерах / Холыднер С. – М. : Бином-Пресс, 2007. – 352 с.
12. Леки-Томпсон, Э. PHP 5 для профессионалов / Леки-Томпсон Э., Коув А., Новицки С., Айде-Гудман Х. – М. : Диалектика, 2006. – 608 с.
13. Веллинг, Л. Разработка Web-приложений с помощью PHP 5 и MySQL 5 / Веллинг Л., Томсон Л. – М. : Вильямс, 2006. – 880 с.

СЕКЦИЯ 7

КОМПЬЮТЕРНАЯ ГРАФИКА
И ДИЗАЙН

ИУС
МКМ 2019

УДК 004.588

Разработка системы «Увлекательная география» с использованием моделей дополненной реальности в межплатформенной среде Unity

М.Б. Бобелюк, Н.Е. Губенко

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: bobeluyk.marina@mail.ru

Аннотация:

М.Б. Бобелюк, Н.Е. Губенко. Разработка системы «Увлекательная география» с использованием моделей дополненной реальности в межплатформенной среде Unity. В докладе рассматриваются технологии и этапы разработки системы «Увлекательная география». Приводится структура и логика приложения, реализация Дополненной реальности, пример работы алгоритма определения ключевых точек на изображении, необходимые условия для создания пакета установки для ОС Android.

Annotation:

Marina Bobelyuk, Natalia Gubenko. Augmented reality as an effective tool in solving the problems of the education system. The article considers the technologies and stages of the development of the system “Interesting geography”. The article contains the structure and logic of the application, the implementation of Augmented Reality, an example of the algorithm for determining key points in the image, the necessary conditions for creating an installation package for the Android OS.

Введение

В последние годы увеличилось количество новых и доступных технологий. Это касается как программ для разработчиков, так и разработанных ими приложений, устройств для обычных пользователей.

На данный момент одним из наиболее перспективных направлений развития образовательных технологий является применение Дополненной реальности. Реализация AR-приложений стала проще, а устройств, которые ее поддерживают – больше.

В данной статье рассматривается разработка системы «Увлекательная география» с использованием молодых, но популярных технологий, доступных для изучения и использования в некоммерческих целях.

Цель статьи

Определение и описание основных понятий, технологий и этапов разработки системы «Увлекательная география».

Интерфейс программы и основные компоненты

Разработка приложения в Unity начинается с создания сцены, которая включает все объекты разрабатываемого уровня приложения.

В стандартном представлении интерфейса программы слева находится панель Hierarchy (Иерархия), она отображает все объекты (GameObject) в текущей сцене; слева – Inspector (Инспектор), он содержит детальную информацию о текущем выбранном объекте, включая все прикрепленные компоненты и их свойства. Здесь можно изменять функционал объектов в сцене. Любое отображенное в Инспекторе свойство можно изменять напрямую из скрипта. Переменную, заданную с модификатором доступа public объектного типа

(GameObject или Transform), можно увидеть в Inspector и назначить на эту переменную объекты в сцене [1].

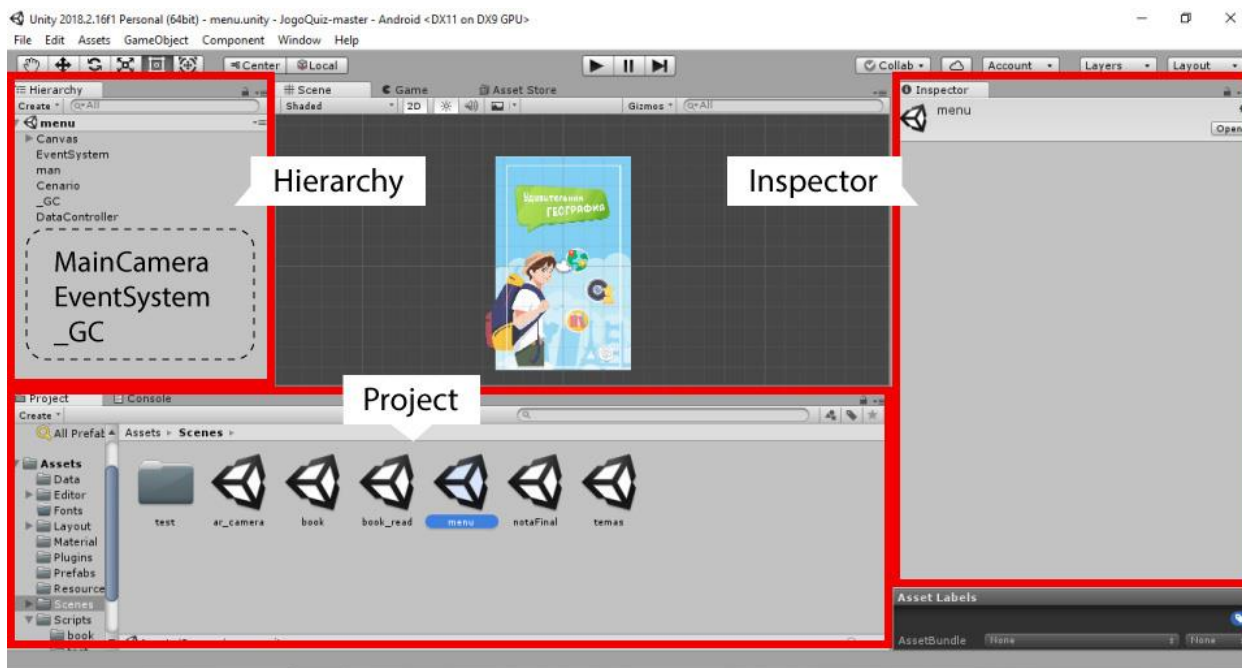


Рис. 1. Интерфейс программы

Каждая сцена содержит объекты MainCamera, EventSystem и _GC. Исключением является сцена с подключением AR-камеры, где объект MainCamera заменен на соответствующий в Vuforia – AR-camera и добавлен объект Image Target.

MainCamera – устройство для показа игрового мира игроку [2].

EventSystem – стандартный объект, ответственный за отправку событий к объектам в приложении.

_GC – объект, который содержит скрипт текущей сцены и представляет доступ ко всем переменным модификаторам доступа public.

Создание скрипта

Создавать скрипт C# можно напрямую из Unity, вызвав соответствующее действие из панели меню или щелкнув правой кнопкой мыши в панели Project и выбрав пункт «Create C# Script». Создается шаблон скрипта с подключенной библиотекой UnityEngine и классом, название которого соответствует названию созданного скрипта. Класс наследуется от класса MonoBehaviour и содержит две пустые функции: Start() и Update().

UnityEngine – стандартная библиотека Unity.

MonoBehaviour – базовый класс, от которого наследуются все скрипты.

Start() – функция инициализации, вызывается перед прорисовкой первого фрейма.

Update() – функция обновления кадра, вызывается каждый фрейм.

Текстовый формат данных

Для чтения информации внутри разрабатываемого приложения был выбран текстовый формат обмена данными JSON, который основан на JavaScript.

JSON (JavaScript Object Notation) – это формат, реализующий неструктурированное текстовое представление структурированных данных, основанное на принципе пар ключ-значение и упорядоченных списках [3].

JSON нацелен в первую очередь на отправку данных между веб-серверами и браузерами или мобильными приложениями. Формат имеет простой дизайн, его легко читать и понимать разработчику, а также интегрировать с большинством языков программирования.

Отсутствие строгой схемы обеспечивает гибкость формата. На официальном сайте Unity приводятся руководства по установке и использованию формата JSON, а также создание дополнительного интерфейса для создания и редактирования файла из панели разработки внутри среды Unity. Приняв во внимание удобство использования JSON, он был выбран для реализации хранения и редактирования информации.

Реализация Дополненной реальности

Для создания дополненной реальности были использованы технологии Vuforia: в сцене задействованы объекты AR-camera и Image Target.

Vuforia – это платформа дополненной реальности и инструментарий разработчика программного обеспечения дополненной реальности (Software Development Kit – SDK) для мобильных устройств, разработанные компанией Qualcomm. Vuforia использует технологии компьютерного зрения, а также отслеживания плоских изображений и простых объемных реальных объектов в реальном времени [4].

AR-camera – объект, связанный с программным модулем Vuforia, ответственным за обнаружение метки и слежение за ней.

Image Target – метка, относительно которой будут размещаться виртуальные предметы. Данный объект управляется стандартными скриптами Vuforia SDK.

Vuforia позволяет создавать маркеры разных видов. В разрабатываемом приложении используются маркеры на основе трекинга особенностей изображения. База изображений для трекинга создается на портале Vuforia Developer. Созданная база скачивается и импортируется в проект.

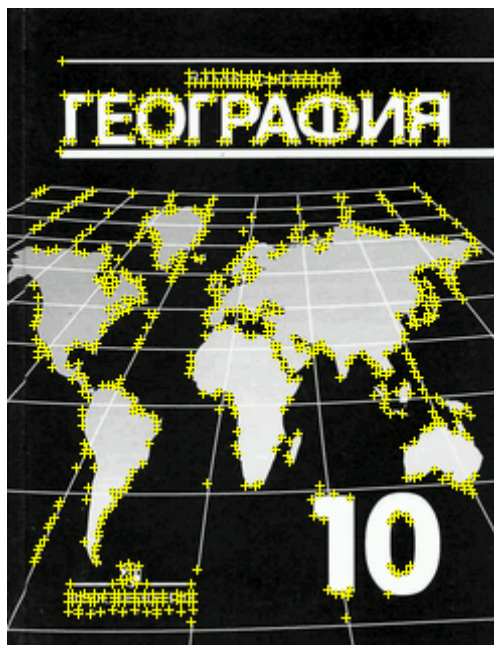


Рис. 2. Определение ключевых точек на изображении

Создание пакета установки для ОС Android

Для создания приложения на ОС Android необходимы подключенные в Unity SDK и JDK. Они позволяют собрать Unity-проект в пакет установки с расширением. apk.

Android SDK – пакет средств разработки программного обеспечения для Android.

JDK – комплект разработчика приложений на языке Java, включающий в себя компилятор Java (javac), стандартные библиотеки классов Java, примеры, документацию, различные утилиты и исполнительную систему Java.

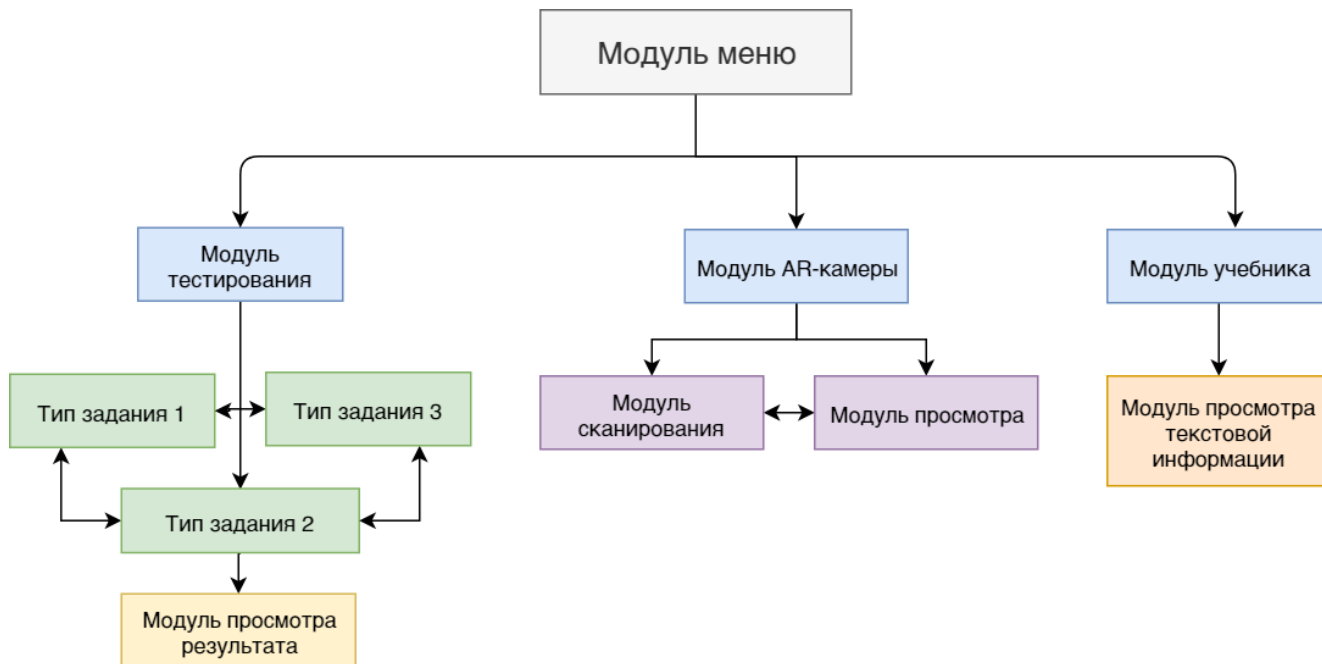


Рис. 3. Структура программы

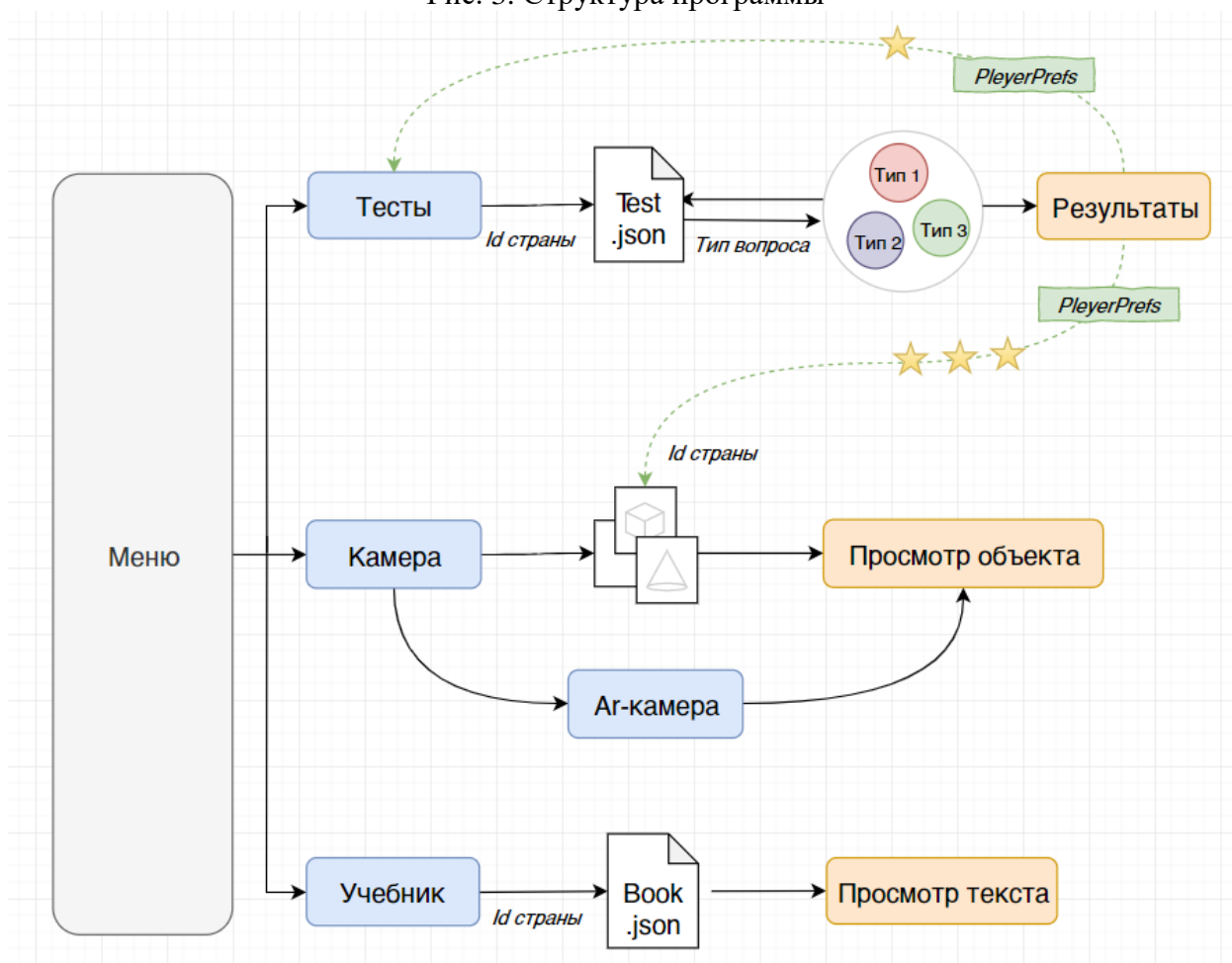


Рис. 4. Логика игрового процесса

Выводы

Для достижения поставленной цели были проанализированы технологии разработки приложений с элементами Дополненной реальности и описаны основные понятия, этапы работы. Разработаны структура приложения и логика игрового процесса, приведен пример работы алгоритма определения ключевых точек на изображении, определены необходимые технологии для создания пакета установки для ОС Android.

В процессе разработки задействованы: межплатформенная среда разработки компьютерных игр Unity, платформа для разработки приложений дополненной реальности Vuforia и редактор исходного кода интегрированной среды разработки Visual Studio 2015.

Литература

1. Srijan Chapagain. Application development with Vuforia and UNITY 3D [Текст] // Thesis. - Centria University of Applied Science, April 2018. - С. 1-43.
2. Unity documentation. Version 2018.3 [Электронный ресурс]. – Режим доступа: <https://docs.unity3d.com/2018.3>.
3. Что такое JSON? [Электронный ресурс]. – Режим доступа: <https://losst.ru/chto-takoe-json>.
4. Vuforia (Software Development Kit) [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/Vuforia>.

UDC 004.05

ANALYSIS OF MODELS AND METHODS FOR STUDYING MULTIMEDIA PRODUCTS CUSTOMER SATISFACTION

A.O. Kandaurova, N.Ye. Gubenko, Ye.N. Kushnirenko

Donetsk National Technical University

E-mail: rafalelka97@gmail.com

Abstract. A.O. Kandaurova, N.Ye. Gubenko, Ye.N. Kushnirenko. Analysis of models and methods for studying multimedia products customer satisfaction. This article discusses the concepts of customer satisfaction and loyalty. Their role in the success of multimedia product developers is determined. The analysis of existing modern methods and models of consumer loyalty research is conducted.

Кандаурова А.О., Губенко Н.Е., Кушниренко Е.Н. Анализ моделей исследования удовлетворенности и потребителей мультимедийной продукции. В данной статье рассматриваются понятия удовлетворенности и лояльности потребителей. Определяется их роль в успешности разработчиков мультимедийной продукции. Проводится анализ существующих современных методик и моделей исследования лояльности потребителей.

The problem statement

Nowadays, in practice, it is not done to predict the level of customer satisfaction of the product being created, which can later become the cause of its “failure” in the market. In organizations engaged in the development of multimedia products, the communication process with the client is often not established well. This is due to the work specifics, because product development is a creative process in many ways that is difficult to document and lead to uniform requirements. This leads to the fact that there are often inaccuracies in work organizing, ambiguity and gross errors on the part of both the developer and the client. The main reason of it is misunderstanding between the client and developer. In turn, it is caused by the lack of information

about the project, distorted by its submission, etc. In this connection, many organizations resort to a strategy aimed at increasing customer satisfaction and loyalty. As part of this strategy, customer satisfaction and loyalty are monitored, customer relationship management systems (CRM) are being introduced, and consumer surveys are being conducted. Ensuring customer satisfaction is one of the key tools for the success of a product being created. To retain existing ones, to replenish the regular customers ranks, to create more efficient products are common tasks for all developers, and their effective solution is impossible without introducing into the managing practice of customer satisfaction continuous monitoring.

This explains the relevance of the chosen topic for organizations interested in creating the monitoring system of customer satisfaction and loyalty and obtaining scientifically based data from consumers for making management decisions of both strategic and tactical plans.

The role of customer satisfaction in the success of developer

Satisfaction is defined as the psychological state of a person as a result of achieving the desired goal. It characterizes the expectations compliance degree from the purchased product with its actual condition. Consumer satisfaction is defined as “a feeling experienced by a consumer after purchasing or using a product” [1]. In well-known sources, it is noted that satisfaction is formed as a result of a comparison of the expected and actual properties of the product and all that is associated with it.

The purpose of the study of customer satisfaction is to ensure that the maximum number of customers' expectations come true. And it would be better if they are “covered” by the product characteristics, causing the customers delight. To manage the satisfaction level efficiently, it is necessary to clarify the mechanism of its formation and the influencing factors. Customer satisfaction measuring is a must-have attribute of customer relationship management systems (CRM).

Before starting the development of a product, it is necessary to find a solution that could reduce the operating time and minimize errors at the developing stage. During solving these problems, special attention should be paid to customer satisfaction, since it is the basis for their loyalty formation to the developer, which later becomes the source of his financial well-being, thereby playing an important role in his success in the market.

Today customer satisfaction is amount to quality management standards ISO 9000:2000, ISO 9001:2000, ISO 9004:2000.

$$\{\text{Quality}\} = \{\text{Customer Satisfaction}\} = \{\text{Value}\} / \{\text{Cost}\}$$

Standards cover the following areas: management responsibility, resource management, product sales, measurement, analysis and improvement.

Evaluation and management of customer satisfaction allows to:

- do better what is meaningful for consumers;
- determine consumer needs and their relative importance;
- understand how consumers perceive the product being created;
- identify unmet needs and areas that are most significant to consumers;
- identify the product extra functions;
- set goals to improve the product and to control the process of achieving them;
- improve performance by increasing customer loyalty.

The analysis of customer satisfaction makes it much easier to compile a general picture of the multimedia product being created, highlight the main requirements for it, and also to evaluate the work of the developer for errors much easier.

The relationship that is established by satisfaction between the developer and the process of multimedia products consumption is expressed through the attitude of the consumer to the product and depends on a specific time, a specific consumer segment, the commodity market and its structure.

Customer satisfaction is influenced by some fundamental factors: the developer established

reputation, goods quality, consumer expectations about the product, using value.

It should be noted that customer satisfaction shapes customer loyalty directly. Consumer loyalty is determined by the positive consumers attitude in point of the entire activity of the developer, its products and services, to the company's staff, its image, brand, logo. Satisfaction and loyalty are closely linked, therefore, only highly satisfied consumers remain loyal to the goods and services of organizations. To date, in connection with this, several universal models of consumer loyalty research have been developed, which work with different efficiencies on different product groups. Therefore, it is advisable to analyze models possible application to assessment of multimedia products consumer loyalty.

Modern models and consumer loyalty research methods

Methods of studying loyalty, as a rule, are divided into two groups. They are empirical methods and mathematical methods. However, they are quite rare. Usually they are used fully, as they allow you to determine different characteristics of the process under study. Empirical methods are aimed at identifying loyalty and determining its level, and mathematical methods are aimed at building a loyalty curve, identifying satisfaction, calculating support index and influence of factors which loyalty is formed through.

There are six main methods of customer loyalty research, among which there are both empirical and mathematical.

The method of needs separating arose in the 50s of XX century based on the experience of many companies engaged in trade and services. Its meaning is to determine the degree of customer loyalty in the numerical equivalent. For example, if a person purchased the product he needed from a conditional company 7 times out of 10, it says that it satisfies 70% of his needs. And the level of loyalty here is determined by how many times a customer turns to the same company, ignoring similar products from other companies. But the method of needs separating has one major drawback notably customers can buy goods of a particular company not only because they are truly loyal. And it is very difficult to determine exactly which purchases were made because of true loyalty. In addition, the percentage of repeated purchases is a very subjective value in order to determine loyal and disloyal customers.

The conversion model was proposed by two researchers, Jan Hofmeyr and Butch Rice. It allows you to determine the degree of customer commitment. The main indicators here are: brand satisfaction, the importance of brand choice, alternatives and fluctuations. Satisfaction indicates that the higher its score, the greater commitment. But satisfaction cannot be correlated with behavior, so it is not possible to disclose the reasons for the customers actions fully. If the brand is not important for the client, then commitment will be difficult to achieve. It means that the brand should be of interest to the person. And if the brand matters to him, he will definitely take the time to decide which one suits him best. Another reason why the client retains loyalty to the company is the understanding that alternatives are worse than the chosen brand. However, if the client can find more interesting alternative for him, he will be able to change his preferences.

The ANA methodology was developed by aircompany «ANA». Exploring customer loyalty issues, they distinguished four main customer segments:

- customers who do not have the choice and are forced to purchase the product of one company;
- unsatisfied and negatively inclined customers;
- fuzzy customers;
- regular customers.

Great importance in the ANA model is given to attracting regular customers from the segment of fuzzy customers. But the company should not just meet the client needs, but “work” beyond his expectations as well, because the partially satisfied client may choose another company, supposing that the new choice will be better. Thus, in order to keep the client, the company must both attract new customers and retain old ones and motivate them to make new acquisitions.

The technique proposed by customer loyalty expert Peter Wilton is interesting because of several levels of loyalty. In particular, the author divides loyal clients into "lawyers" and "allies" of the company. Lawyers are people who treat the brand positively, are faithful to it and give recommendations to other people. But the company ultimate goal should be to create "allies" who not only committed to the company and satisfied with it, but also participate in business processes.

The author of another method is a well-known American marketing specialist David Aaker. It offers several options for measuring loyalty based on research: consumer behavior, costs accounting, satisfaction, relationship to the brand, commitment.

The easiest way to measure loyalty is to observe the buyer's behavior. The key parameters here are:

- repeat purchases indicators;
- purchases percentage (how many purchases of the five fall on each brand);
- the number of purchased goods of different brands (how many customers purchased goods of one brand, how many – two, etc.).

Changing costs imply studying the preferences of customers in purchasing goods of one brand and their unwillingness to change their preferences. The degree of satisfaction is a fundamental factor, and the attitude to the brand can bet in respect, trust, willingness to pay a price that exceeds the price of competitors, etc.

Nowadays, the Aaker technique has found wide application. In order to implement it, various types of surveys and questionnaires are used to determine each specific indicator of loyalty, for example, a relationship or level of satisfaction. Then loyalty indices are calculated by the arithmetic values of average points for all aspects. Thus, it is possible to make an estimation of all factors that form the loyalty degree.

Another model was created by Jean-Jacques Lambin who is one of the most famous representatives of the European marketing school. His loyalty estimation is carried out by three criteria: the quality of the product main benefits, providing services process, services value perceived by the client.

The company that manages loyalty should estimate changes in customer commitment, comparing it with the greatest leaps periods, when profits increased owing to new customers influx and repeated purchases. The company should also solve new problems and find new directions for its development.

Analyzing the presented methods of loyalty research, it can be concluded that with their complex application all the necessary indicators can be determined: the loyal availability, its level and development trends, and also to analyze the satisfaction with the product and the company service.

All this information helps to find out how it is necessary to set the further loyalty management process, to understand which changes occur in loyalty over time and to find ways to improve the service quality, which will take into account the consumers interests and features. To solve this problem, the system is being developed, which Use Case Diagram is shown in figure 1.

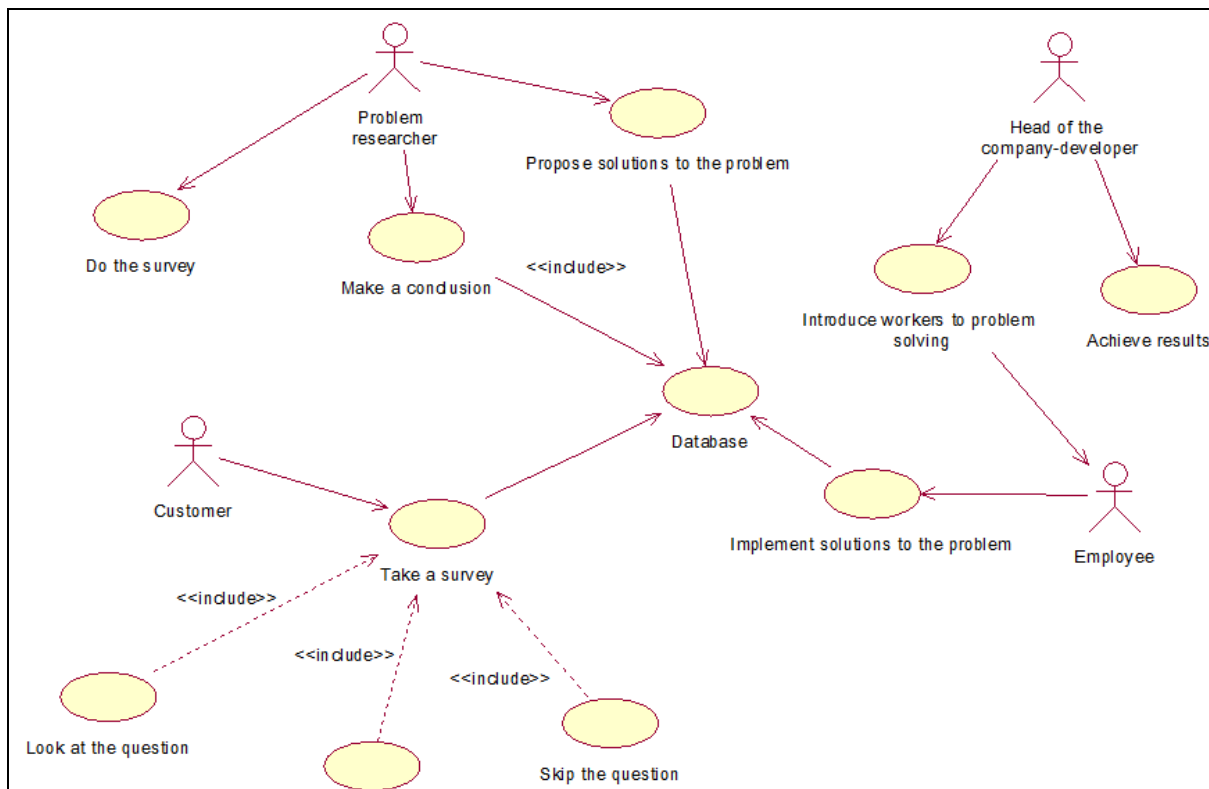


Figure 1 – Use Case Diagram

Conclusion

In this way, the study of satisfaction and loyalty of multimedia products customers is an important element in developers work. Since, relying on the consumers wishes, the organization will be able to fulfil their requirements in full and anticipate their expectations, thereby increasing the quality of the products created. It will allow to achieve not only financial success but consumer commitment as well. Consequently, there arises the expediency to create a system for analyzing customer satisfaction and loyalty in order to implement the strategy of the multimedia products developing company. Research of consumer loyalty methods and models was carried out. During researching, it was revealed that using of these models and methods, it is possible to determine all necessary indicators for customer satisfaction study with the company's product and service. The Use Case Diagram of the developed system for analyzing the customer satisfaction level is made.

References

1. Салимова Т.А. Управление качеством. – 2-е издание, М.: Омега –Л, 2008. – 416с.
2. Исследование лояльности клиентов [Электронный ресурс] // 4Brain, 2018: [сайт] – Режим доступа: <https://4brain.ru/blog/> – Загл. с экрана.
3. Полынская, Г. А. Сравнение методов оценки удовлетворенности потребителей при использовании разных способов сбора данных / Г. А. Полынская, // Управление экономическими системами. – 2014.
4. Турко С.В. Автореферат диссертации на соискание ученой степени кандидата экономических наук "Мониторинг удовлетворенности и лояльности потребителей // Кафедра стратегического управления и развития человеческих ресурсов Московского государственного университета экономики, статистики и информатики – Москва, 2006.

УДК 006.015.5

МЕТОДЫ АНАЛИЗА УДОВЛЕТВОРЕННОСТИ ПОТРЕБИТЕЛЕЙ РЕКЛАМНЫХ УСЛУГ

Кандаурова А.О., Губенко Н.Е.

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: rafalelka97@gmail.com

Аннотация:

Кандаурова А.О., Губенко Н.Е. Методы анализа удовлетворенности потребителей рекламных услуг. В статье представлены основные методы анализа удовлетворенности потребителей рекламных услуг.

Annotation:

Kandaurova A.O., Gubenko N.E. Methods of analyzing advertising services customers satisfaction. The article presents the main methods for analyzing the advertising services consumers satisfaction.

Постановка проблемы

Важным фактором в развитии рынка услуг, в том числе рекламных, является удовлетворенность потребителей. Существует множество теорий и методов, которые дают представление о понятии «удовлетворенность» и возможностях её анализа.

Организации, представляющие сферу услуг, достаточно мобильно реагируют на требования потребителей, изучают свойства товаров, проектируют услуги с более высокой дополнительной воспринимаемой ценностью [1]. Такие рекламные услуги, несомненно, будут пользоваться наибольшим спросом. В свою очередь на качество услуг влияет и стратегия грамотно направленная на повышение удовлетворенности и лояльности потребителей. Задача состоит в том, чтобы определить, насколько предложение рекламной компании соответствует или превосходит ожидания потребителей. На основании оценки удовлетворенности компания определяет, какие стороны деятельности необходимо улучшить.

У каждого потребителя присутствуют явные и скрытые потребности и желания, которые оказывают непосредственное влияние на удовлетворенность и формируют уровень лояльности к товару (услуге). Сегодня оценка уровня удовлетворенности клиента стала распространенной практикой для большинства компаний, осуществляющих деятельность в сфере рекламных услуг. Компании регулярно проводят опросы для измерения уровня удовлетворенности покупателей. Существует несколько эффективных методов оценки удовлетворенности.

Балльная оценка удовлетворенности

Подобный метод оценки применяется в ходе опросов потребителей, которых просят оценить уровень удовлетворенности в баллах. Как правило, используется пятибалльная шкала, представленная в таблице 1.

Таблица 1 – Шкала оценки удовлетворенности потребителя

1	2	3	4	5
Абсолютно неудовлетворен	Частично неудовлетворен	Ни удовлетворен, ни разочарован	Частично удовлетворен	Удовлетворен полностью

Рейтинги удовлетворенности могут использоваться для оценки качества предоставленных услуг и обслуживания в целом. Их снижение указывает на наличие проблемы, которая может повлиять на объем продаж и рентабельность. Важны показатели в крайних точках, поскольку, клиенты, которые оценили уровень своей удовлетворенности на 5, с большой вероятностью совершат повторный заказ, а абсолютно неудовлетворенные клиенты вряд ли снова обратятся в компанию.

Мультиатрибутивная модель товара

Удовлетворенность потребителя формируется в результате комплексного воздействия нескольких факторов, поэтому для ее оценки используют известную мультиатрибутивную модель товара М. Фишбеина. В этом случае удовлетворенность определяется как средневзвешенная оценка исполнения или присутствия свойств товара и их важности для потребителя. Оценка свойств товара производится в баллах. Клиентов просят оценить степень удовлетворенности по 10-балльной шкале. А также определить значимость каждого из свойств товара.

Поскольку модель М. Фишбеина является компенсаторной, т.е. низкие показатели по одним атрибутам компенсируются повышенными значениями других, это позволяет одним, более выраженным свойствам компенсировать другие, оценивающиеся более низко. Такой способ оценки не всегда является эффективным. К примеру, если потребитель сталкивается с абсолютным ограничением по цене, оценка уже не имеет компенсаторного характера, так как один критерий доминирует над остальными.

Кроме оценки отдельных атрибутов товаров и удовлетворенности брендом в целом очень важно провести анализ соотношений удовлетворенность/важность. Такой анализ дает очень полезную информацию для принятия управленческих решений (рис. 1).



Рис.1 – Профиль удовлетворенности клиентов

Соотношение удовлетворенность/важность определяет четыре возможные зоны:

- зона 1 – удовлетворенность ниже, чем важность атрибута; задача состоит в повышении удовлетворенности потребителя этим атрибутом путем устранения недостатков;
- зона 2 – хорошая удовлетворенность по важным для потребителя атрибутам; идеальная зона, поскольку важность и удовлетворенность совпадают;
- зона 3 – низкая удовлетворенность по неважным для покупателя атрибутам; эту зону не следует рассматривать как зону для улучшений, поскольку низкий уровень удовлетворенности соответствует низкой оценке атрибута потребителями;
- зона 4 – удовлетворенность существенно выше, чем значимость, придаваемая респондентами данному атрибуту, что свидетельствует о том, что средства, затрачиваемые компанией на создание и поддержание данного атрибута, излишни.

Индекс удовлетворенности клиентов (Customer Satisfaction Index, CSI)

Индекс удовлетворенности потребителей (Customer satisfaction Index, CSI) – это оценка уровня удовлетворенности клиента после взаимодействия с компанией [3]. CSI отражает насколько успешным был клиентский опыт потребителя. Метод основан на опросе потребителей, совершивших то или иное действие в отношении товара, марки или их владельца. Опрос может осуществляться в магазине после покупки, по телефону, если покупатель оставил координаты, по электронной почте и любыми другими доступными методами опроса.

Оценка удовлетворенности включает в себя четыре основных шага:

- первый шаг: произвольно и экспертно определяются параметры, которые компания считает важными, чаще всего оценивают отношение покупателей к товару или услуге, клиентскому сервису, к качеству и количеству информационно-маркетингового воздействия, соответствие цен с потребительским ожиданием и сравнение цен с ценами конкурентов;
- второй шаг: в результате опроса потребителей, выявляется их отношение к этим параметрам по пятибалльной шкале: один балл – неудовлетворенность, пять – удовлетворенность.
- третий шаг: выяснение у покупателей, насколько важен для них тот или иной параметр из экспертного списка по пятибалльной шкале;
- четвертый шаг: анализ полученных сведений об удовлетворенности потребителей.



Рис. 2 – Модель измерения индекса удовлетворенности потребителей

Чистый индекс промоутера (Net Promoter Score, NPS)

Индекс NPS распространен как отдельный инструмент измерения лояльности, так и в комплексе с другими методами. Суть метода в том, что потребителям задаются два вопроса: закрытый и открытый. Закрытый вопрос формулируется следующим образом: "В какой степени вы готовы рекомендовать данную компанию своим друзьям, коллегам, знакомым?" Степень готовности измеряется по 11-балльной шкале, где 10 – потребитель будет с большой долей вероятности рекомендовать компанию, 0 – ни в коем случае не будет рекомендовать, 5 – середина шкалы, означает нейтральное отношение к компании (рис. 3).

В результате опроса клиенты делятся на три группы:

- промоутеры – поставившие оценки 9 и 10 баллов;
- нейтралы – поставившие оценки 7 и 8 баллов;
- критики – давшие оценки от 0 до 6, не удовлетворены компанией и не будут ее рекомендовать.

Следовательно, индекс NPS характеризуется разностью между долей промоутеров и долей критиков.

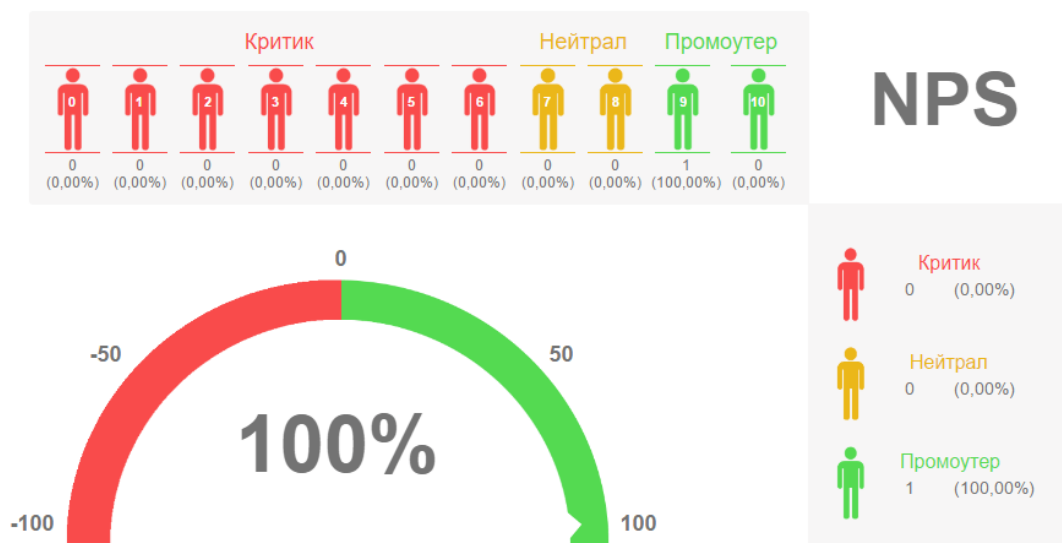


Рис. 3 – Модель вычисления чистого индекса промоутера

Метод Н. Кано

Данный метод используется для оценки эмоциональной реакции потребителей на отдельные характеристики продукции. Полученные с его помощью результаты позволяют управлять удовлетворенностью и лояльностью потребителей. Японский ученый и консультант Нориаки Кано подверг критике общепринятые теории о том, что сохранять лояльность потребителей необходимо путем реагирования на их жалобы и расширения наиболее популярных характеристик продукта.

Кано отталкивался от того, что лояльность потребителей взаимосвязана с эмоциональной реакцией на характеристики предоставляемых услуг (товаров) и выделил пять различных реакций:

- привлекательные характеристики (если они присутствуют в продукте) вызывают чувства удовлетворения и восторга, однако если этих характеристик нет, то пользователи не испытывают неудовлетворения;
- одномерные характеристики вызывают удовлетворение (если они есть) или неудовлетворение (если их нет);
- обязательные характеристики относятся к группе тех качеств, которые обязательно должны присутствовать в продукте; усиление обязательных характеристик постепенно приводит к замедлению роста эмоциональной реакции;
- неважные характеристики вызывают неоднозначную реакцию пользователей, но, в целом, им все равно, присутствуют такие характеристики в продукте или нет;
- нежелательные характеристики сводят на нет положительное влияние привлекательных и одномерных характеристик.

Кано был приверженцем того, что [удовлетворенность потребителей](#) меняется с течением времени. Характеристики продукции, вызывающие восторг в данный момент времени, вскоре перейдут в категорию качеств ожидаемых или требуемых клиентом. Кривая привлекательных характеристик постепенно превращается в кривую обязательных характеристик. Кроме того, само понятие «лучший в категории» постоянно меняется, что влияет на параметры конкретной характеристики на оси X.

В рамках метода Кано предполагается использование анкеты, которая нацелена на выявление характеристик товара (услуги), содержащая по каждому свойству, один из которых касается функциональной стороны, а другой – дисфункциональной. В результате обработки ответов респондентов полученные результаты распределяем в соответствии с представленной ниже таблицей.

Таблица 2 – Ответы респондентов по методу Кано

Функциональная характеристика	Дисфункциональная характеристика				
	Доволен	Обязательно должно быть	Безразлично	Относительно не доволен (терпимо)	Не доволен
Доволен	Свойства, вызывающие сомнения	Привлекательные свойства	Привлекательные свойства	Привлекательные свойства	Одномерная
Обязательно должно быть	Свойства обратного действия	Незначительные	Незначительные	Незначительные	Обязательные свойства
Безразлично	Свойства обратного действия	Незначительные	Незначительные	Незначительные	Обязательные свойства
Относительно не доволен (терпимо)	Свойства обратного действия	Незначительные	Незначительные	Незначительные	Обязательные свойства
Не доволен	Свойства обратного действия	Свойства обратного действия	Свойства обратного действия	Свойства обратного действия	Свойства, вызывающие сомнения

Следующим этапом исследования удовлетворенности в соответствии с методом Кано является определение перечня требований целевой аудитории к каждому свойству по опрашиваемым респондентам, ответы фиксируются в процентном соотношении и заносятся в таблицу.

Результаты исследования Кано позволяют выделить типы потребителей. Дальнейшее разделение результатов по группам типажей показывает, как разные группы реагируют на те или иные характеристики. Здесь необходимо пересмотреть те качества продукта, которые не понравились группе «принимают сразу».

Метод Кано показывает, почему лояльность клиентов нельзя получить и сохранить лишь за счет улучшения существующих характеристик, а также позволяют оценить эффективность вложений в их улучшение.

Выводы

Анализ удовлетворенности потребителей крайне важен для любой организации, связанной со сферой предоставления услуг, в частности рекламных. На основе этих данных можно грамотно построить стратегию развития. Удовлетворенность потребителей должна быть основой любого бизнеса и учитываться при принятии решений. Таким образом, опираясь на желания потребителей, организация будет способна выполнять их требования в полной мере и предвосхищать их ожидания, точнее, тем самым повышать качество предоставляемых услуг, что позволит достичь не только финансового успеха, но и потребительской приверженности.

Литература

1. Рудакова О.Ю. Развитие институциональной модели управления качеством бытовых услуг: автореф. дис. канд. экон. наук: 08.00.05. Шахты, 2012.
2. Способы оценки клиентского опыта. [Electronic resource] / Интернет-ресурс. Режим доступа: <http://fdfgroup.ru/poleznaya-informatsiya/stati/9-sposobov-otsenki-klientskogo-opyta/>
3. Оценка удовлетворенности и лояльности клиентов. [Electronic resource] / Интернет-ресурс. Режим доступа: <https://mtp-global.com/customer-satisfaction-index-2/>
4. Объяснение модели Кано. [Electronic resource] / Интернет-ресурс. Режим доступа: <httpgenerator.ru/blog/2018/07/03/obyasnenie-modeli-kano-analiz-i-primery/>

УДК 006.015.5

АНАЛИЗ ДИЗАЙН ИНДУСТРИИ И БИЗНЕС-ПРОЦЕССОВ В ДИЗАЙН-СТУДИЯХ

Тилинина Н. Ю., Губенко Н.Е.

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: natasha_tilinina@mail.ru

Аннотация:

Тилинина Н.Ю., Губенко Н.Е. Анализ дизайн индустрии и бизнес-процессов в дизайн-студиях. В статье представлены анализ отрасли в которой функционируют компании, занимающиеся дизайном, их внешняя среда и бизнес-процессы.

Annotation:

Tilinina N.Yu., Gubenko N.E. Methods of analyzing advertising services customers satisfaction. The article presents an analysis of the industry in which design companies, their external environment and business processes operate.

Постановка проблемы

В настоящее время дизайн занимает довольно важную роль в жизни каждого, а в частности в становлении на рынке любой компании или производства. Не смотря на это у дизайн-студий так же есть особенности в бизнес-процессах, грамотное следование которым необходимо для комфортного существования на современном рынке. Для этого необходим качественный анализ процессов для выявления проблемных участков.

Общий анализ

Фундамент любого digital-бизнеса — это люди. Точнее — их позиции и функции. Чтобы снизить зависимость от личности и максимально стандартизировать работу, избегая ошибок, у вас обязательно должны быть внедрены бизнес-процессы.

Дизайн-индустрия имеет свои особенности, отличающие её от других отраслей. По данным отчёта Barnes [4], ежегодного издания, которое публикует статистику по различным индустриям в 60ти странах мира, индустрия графического дизайна в России насчитывает больше девяти с половиной тысяч предприятий. В 2011 году наблюдался скачок этого показателя- появление большого количества новых студий, а в 12м году был резкий спад численности предприятий, после чего рынок «устаканился», и по сей день стабильно растёт, но к показателю 2011 года по количеству компаний до сих пор не вернулся. В этом же отчёте сообщается, что показатель продаж в отрасли постоянно увеличивается.

В среднем количество сотрудников дизайн-студии составляет 4 человека, при этом каждое предприятие зарабатывает около 120 тысяч долларов в год . Коэффициент трудоёмкости в отрасли низкий – используется интеллектуальный труд, а продуктивность сотрудников очень высокая (один из основных ресурсов – это люди и их интеллектуальные способности).

Проанализируем среду, в которой находятся дизайн-компании по факторам – сложность, стабильность, враждебность и разнообразие рынка, описанных в предыдущих главах. Окружение дизайн-студии имеет определённую динамичность, на которую компании приходится постоянно реагировать, что способствует децентрализации. Вместе с динамичностью, внешняя среда характеризуется сложностью, так как чаще всего продукт, производимый дизайн-студией уникален и на производство одной его единицы тратится большое количество ресурсов, в процессе его создания приходится выполнять множество видов творческой работы, что способствует уже децентрализации компании и гибкости.

Интегрированность рынка будет зависеть конкретно от специализации каждой студии, от того, специализируется она на чём-то одном (например, на логотипах) или занимается многими видами деятельности (сайтами, рекламными компаниями и брендингом). Чем большим количеством видов деятельности занимается дизайн-студия, тем больше как правило её штат и вероятнее, что она будет иметь организационную структуру с несколькими отделами. Враждебностью данная среда обладает небольшой.

Поэтому исходя из типов организационной структуры по Г. Минцбергу дизайн-студии в общем будут обладать децентрализованной структурой с органическим типом управления и коммуникацией на основе взаимного согласования.

Исходя из модели жизненного цикла отрасли М. Портера, можно сказать, что дизайн индустрия в России находится на стадии становления, технологии функционирования компаний постепенно налаживаются, при том что многие покупатели до конца не понимают за каким продуктом они приходят в дизайн агентства и как выстраивается сотрудничество. На западе ситуация совсем другая, там эта отрасль находится в стадии зрелости, потому как основное её развитие пришлось на годы, когда в нашей стране советский режим не нуждался в дизайн индустрии.

Рассмотрим дизайн-индустрию с точки зрения пяти конкурентных сил, влияющих на компании:

– Угроза входа новых игроков велика, так как низок барьер входа в отрасль. Начальный капитал – небольшой, достаточно персонального компьютера и человека, который умеет пользоваться специализированным программным обеспечением и обладает специальными знаниями. По сути фирма действительно может состоять из одного специалиста. Доступ к сбытовой цепи легко осуществляется через интернет. Единственной проблемой на начальном этапе является отсутствие репутации, но это решается наработкой качественного портфолио.

– Угроза поставщиков маленькая, потому что поставщиками для дизайн-студий являются компании, продающие ПК, интернет провайдеры, типографии – их много и они обеспечивают сервисом множество отраслей. Единственные специализированные поставщики, оказывающие особое влияние на отрасль – это фирмы, которые производят специальное программное обеспечение для дизайнеров.

– Угроза покупателей также не оказывает существенного влияния на отрасль, потому что покупателей больше, чем производителей.

– Угрозу со стороны субститутов дизайн отрасли сложно определить. Если клиенту необходимы услуги дизайнера то фирмами, производящими субституты, можно считать стоки готовых логотипов, которые не предоставляют индивидуального обслуживания конкретного бизнеса, зато имеют привлекательную цену. В случае же если клиент хочет использовать дизайн для повышения продаж, в качестве субститута может выступать реклама, услуги маркетолога или бизнес-аналитика.

– Конкуренция в отрасли высока но, не ожесточённая. Потому что рынок дизайн услуг поделён на сегменты и с точки зрения видов услуг и по цене этих услуг, имеет несколько целевых аудиторий, на которые ориентируются компании от массового спроса и производства дешевой не очень качественной продукции новичками, которая находит своего потребителя, до комплексного обслуживания корпораций супер профессиональными командами дизайнерских агентств. То есть число игроков на рынке велико, но они имеют абсолютно различные сферы влияния, отрасль растёт.

Управление коммуникациями дизайн-студии

Итак клиенты указали, что «умение общаться» - наиболее важное качество для сотрудников дизайн-студии. Общаться разными способами – очно, по телефону, через соцсети или электронную почту. А еще необходимо время, что бы разработать эффективную стратегию коммуникации с каждым клиентом. Лидеры отрасли от остальных компаний

отличаются тем, что эта составляющая у них отработана до совершенства. Это достигается прежде всего – подбором кадров. В таком деле мало обладать личным обаянием – коммуникацию с каждым клиентом необходимо заключить в рамки и сроки.

В арсенале бизнес-технологий есть большой набор средств для оптимизации, ускорения, упрощения коммуникаций. Конечно, когда заказчиков мало, 1-2, формализованные системы взаимодействия с клиентами не нужны. Но как только мини-студия набирает даже небольшой портфель заказов, коммуникации начинают давать сбой в самых неожиданных местах. Теряются правки заказчика, вовремя не доводятся точные сроки, и так далее.

В других странах эту проблему решают с помощью автоматизации процесса – применяют CRM (Customer Relationship Management) – системы, являющиеся своего рода «отражателями» рабочего процесса.

Управление финансами

Финансы – это важная сторона любого бизнеса, но что касается творческих организаций, здесь все немного запутано. Есть мнение, что творчество должно приносить удовольствие, и этого уже достаточно, поэтому периоды, когда нет клиентов, когда стоимость заказов надо снижать из-за их недостатка, надо переносить стоически. Настоящие профессионалы же совмещают удовольствие и прагматичный подход – выстраивают стратегии развития, подключая современные методы управления, технологии, финансовое планирование.

Все что зарабатывает творческая студия обычно идет на оплату труда персонала и на самопродвижение. К тому же надо учитывать периоды «спячки» - когда 2-4 месяца нет заказов. На этот случай необходим специальный фонд – «заначка».

Ценовая модель

Оценка стоимости работ, выполненных дизайн –студией серьезно различается у нас и за рубежом, как по методологии, так и в ценовом диапазоне. У нас на один и тот же вид работы цены могут отличаться значительно.

Российские дизайнеры предпочитают рассчитывать цену в зависимости от приблизительной оценки сложности проекта. Поэтому заказчики зачастую сталкиваются с большим разбросом цен в рамках одного тендера.

В таких странах, как Великобритания и США, существует исторически сложившаяся для творческой работы модель расчета стоимости в зависимости от потраченных на неё часов.

Развитие студии

Она из наиболее распространенных ошибок дизайн-студий, начинающих свою работу: когда через несколько лет успешной деятельности фирма набила руку и набрала устойчивый поток клиентов., владельцу начинает казаться, что нет развития, надо куда-то дальше двигаться. Тогда принимается решение «заняться всем» - и теряется экспертность в отдельно взятой области, где эта студия считалась асом. В литературе это называется – потеря стратегического фокуса и уход от уникального позиционирования.

Однако, это не значит, что развивать новые виды деятельности – это плохо. Ведь созданы начальные условия для развития новых креативных услуг.

Прогнозы

Специалисты считают, что будущее творческих студий лежит в плоскости объединения разных видов деятельности для создания на их базе чего-то нового. Дизайн-исследования, обзор трендов развития, аудит дизайна. Подобные виды комплексных услуг возможны при слиянии компаний разной специализации.

Выводы

В данной статье была проанализирована окружающая среда дизайн-компаний, бизнес-процессы, которые участвуют в полноценном функционировании дизайн студий, и

составлена характеристика дизайн индустрии в России, которая согласно жизненному циклу отрасли М.Портера находится на стадии становления.

Литература

1. Минцберг Г. Структура в кулаке: создание эффективной организации; пер. с англ.- Спб.; Питер. 2004 – 512 с.
2. Минцберг Г., Куин Д.Б., Гошал С. Стратегический процесс. Концепции. Проблемы. Решения. – Спб.; Питер. 2001 – 688 с.
3. Портер М. Конкурентная стратегия. Методика анализа отраслей конкурентов. – М.: Альпина Паблишер. 2011 – 454с.
4. С. Barnes & Co Worldwide Graphic Designs Services Industry-Industry Market report// Barnes Reports, 2015

УДК 004.588

РАЗРАБОТКА ИГРЫ «ЛОГИЧЕСКИЙ БАСКЕТБОЛ»

Лашенова В.Э., Киселева О.В.

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: lerchik71@gmail.com, olgakiseleva_donntu@mail.ru

Аннотация:

Лашенова В.Э., Киселева О.В. Разработка игры «Логический баскетбол». Рассмотрен процесс проектирования игры от определения назначения игры до ее тестирования.

Annotation:

Lashchenova V.E., Kiseleva O.V. Development of the game "Logical Basketball". The process of designing a game from determining the purpose of the game to testing it is considered.

Введение

Обучение – непрерывный процесс в жизни каждого человека. И чем раньше он начнется, тем больше шансов развить или укрепить в себе какие-либо навыки. В современном мире обучение стараются начать как можно раньше, особенно при подготовке ребенка к школьному процессу приобретения знаний. Игра «Логический баскетбол» помогает ребенку младшего школьного возраста тренироваться в упражнениях на логическое мышление и представляет собой тренажер.

Разработка игры

После того, как была определена цель игры – развитие логического мышления и целевая аудитория – дети младшего школьного возраста, был спроектирован сценарий игры, который предполагает то, что дети должны прочесть вопрос на экране, понять правильный ответ и забросить мяч в корзину, над которой висит правильный ответ. Бросок мяча осуществляется щелчком ЛКМ по мячу и движением мышки по траектории экрана, пока не наведем на нужную область кольца, чтобы в него попасть. Всего попыток три, если игрок не попадает в нужное кольцо все три раза, то переходим на следующий уровень и очко игроку не засчитывается. В случае, если игрок попадает все три раза в ошибочное кольцо, то также переходим на следующий уровень без дополнительных очков. Всего таких вопросов 15, и после того, как игрок ответил на все появляется результирующее меню, в котором

отображаются все ответы игрока. Причем, если ответ был правильный, то он подсвечивается зеленым цветом, если нет, то красным.

Вторым шагом в разработке игры стало разработка интерфейса в программе Photoshop CC 2018 и проектирование моделей баскетбольного мяча и кольца в программе Blender (рис. 1).



Рис. 1. Модель баскетбольного кольца

После разработки самой модели в программе Unity был наложен материал на кольцо и баскетбольный мяч.

Третьим шагом, в разработке игры, стало моделирование двух сцен – Меню и Уровни.

Меню отвечает за стартовое меню и за окно с результатами, а уровни – соответственно за сам игровой процесс. В сцене «Уровни» был определен массив вопросов и всех видов ответов, как правильных, так и не правильных (рис. 2.).

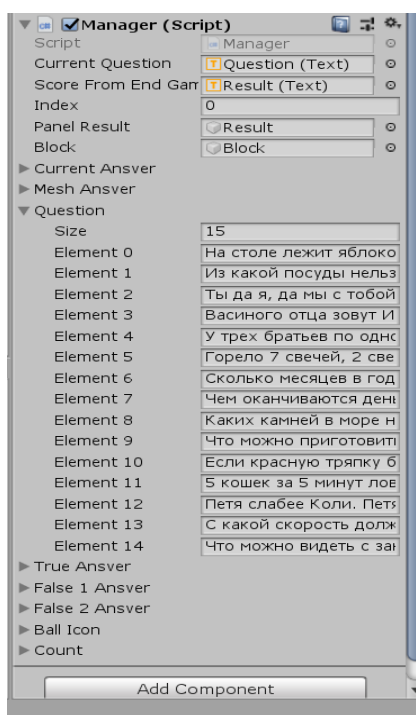


Рис. 2. Вопросы и ответы

После этого был написан программный код для каждой группы элементов.

Завершающим шагом стало тестирование игры. Тестирование проводилось при помощи метода «белого ящика», так как исходный код был известен заранее. После выявления всех ошибок, было проведено их исправление.

Выводы

В результате проделанной работы была разработана игра «Логический баскетбол» для детей младшего школьного возраста, которая позволяет развивать логическое мышление при помощи несложной игры.

Игровые уровни представлены на рисунках 3 и 4.

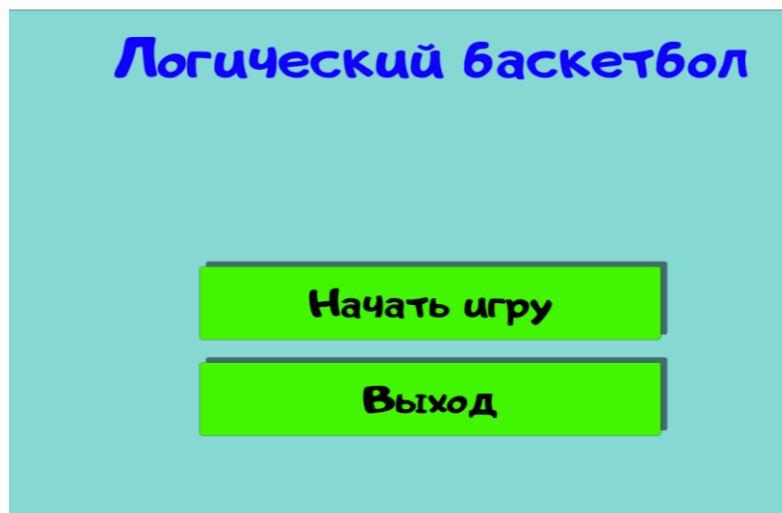


Рис. 3. Меню

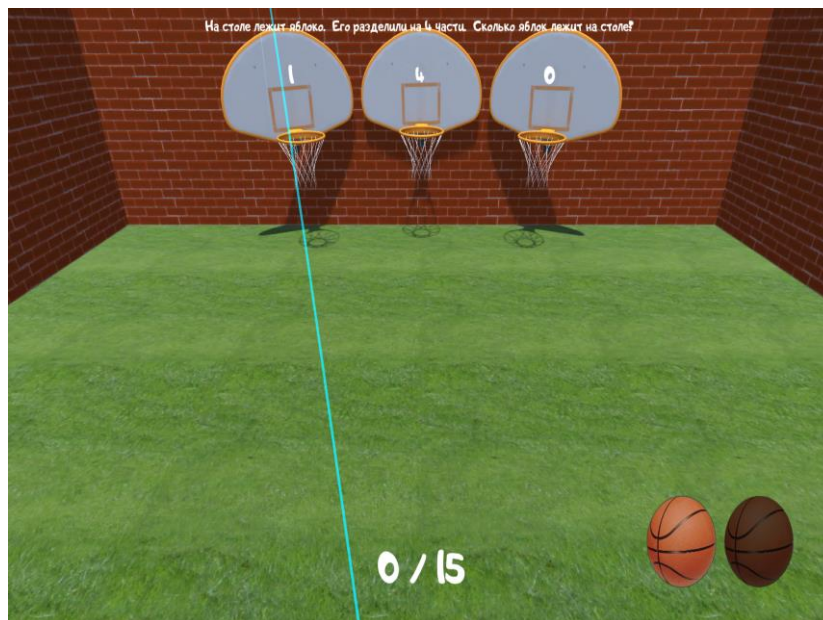


Рис. 4. Уровень 1

Литература

1. Unity3d. Начало работы, практические советы. / Интернет-ресурс. Режим доступа: <https://habr.com/ru/post/161463/> - Загл. с экрана.
2. Создаем игру с нуля. / Интернет-ресурс. Режим доступа: https://pikabu.ru/story/sozdaem_igru_s_nulya_v_unity3d_3519138 - Загл. с экрана.

УДК 003.026

ПОВЫШЕНИЕ ЗАЩИЩЕННОСТИ КОНФИДЕНЦИАЛЬНОЙ ИНФОРМАЦИИ НА ОСНОВЕ КОМБИНАЦИИ МЕТОДОВ ЖАРГОНОВ

Сидорчук В.И., Губенко Н.Е.

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: nohohone@gmail.com

Аннотация:

Сидорчук В.И., Губенко Н.Е. Повышение защищенности конфиденциальной информации на основе комбинации методов жаргонов. Рассмотрено понятие информации. Рассмотрены классификации информации. Была выявлена цель защиты конфиденциальной информации. Был рассмотрен один из методов защиты информации. Был предложен авторский метод шифрования информации, при использовании которого может повыситься уровень информационной безопасности.

Annotation:

Sidorchuk V.I., Gubenko N.E. Improving the security of confidential information based on a combination of jargon methods. The concept of information is considered. Considered the classification of information. The purpose of protecting confidential information has been identified. Was considered one of the methods of information security. A author's method of information encryption was proposed, which can increase the level of information security.

Постановка проблемы

Информацией принято считать любой результат работы человека (идея, технология, программа, различные данные), который не является физическим предметом и может быть использовано человеком. Не каждая информация может находиться в свободном доступе. Существуют несколько видов информации, которые требуют конфиденциальности. Основной целью защиты конфиденциальной информации состоит в том, чтобы предотвратить ее утечку и овладение этой информацией сторонними лицами.

С развитием информационного общества проблемы, связанные с защитой конфиденциальной информации, приобретают всё большее значение. Одним из решений является повышение защищенности конфиденциальной информации путем использования комбинации методов жаргонов.

Исследования

Актуальность исследования данной темы обусловлена тем, что информация является одним из важнейших продуктов общественного производства, постоянно наращиваемый ресурс человечества. На международном уровне сформировалась система взглядов на информацию как на ценнейший ресурс жизнеобеспечения общества, имеющий социальное значение.

В предпринимательской деятельности (или любой другой) может использоваться разнообразная информация. Вся используемая информация представляет собою различную ценность для субъекта и, соответственно, ее разглашение может привести к угрозам экономической и интеллектуальной безопасности. В связи с этим, информация делится на два типа: информация открытого доступа (может быть использована любым лицом) и информация ограниченного доступа (для пользователей имеющих соответствующие права). На рисунке 1 изображена более подробная структура типов информации.

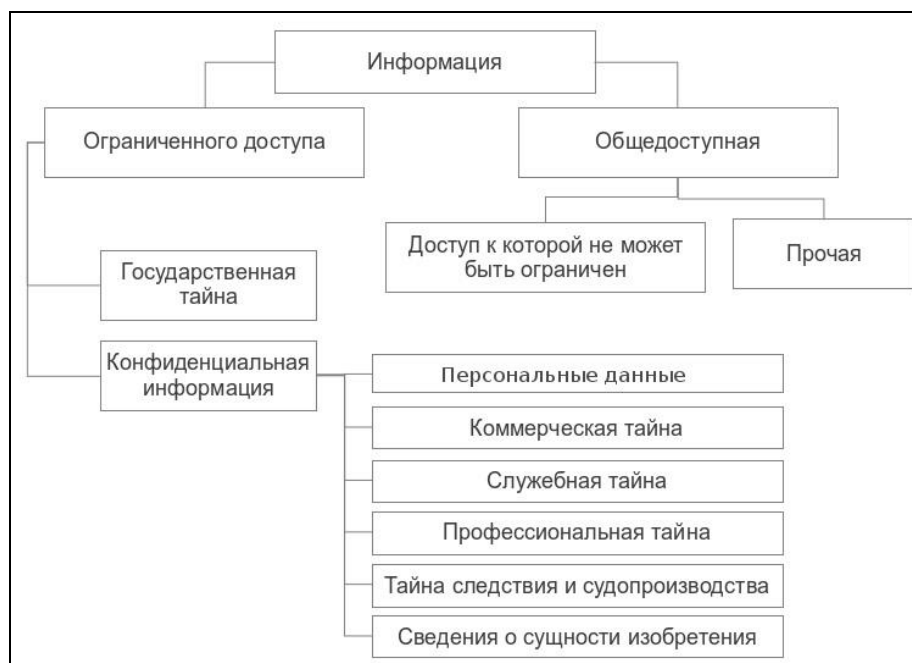


Рисунок 1 – Классификация информации

На сегодняшний день целесообразность использования разных информационных технологий влечет за собой угрозу информационной безопасности и утечке «закрытой» информации.

Потеря информации на персональном компьютере, или же перехват документ при отправке электронного письма, ощутима для его владельца и может привести различного рода проблемам. Проблема защиты конфиденциальности информации носят глобальный характер. Методы реализации угроз информационной безопасности постоянно совершенствуются, высокая технологичность этих угроз требует адекватных мер противодействий (рисунок 2).

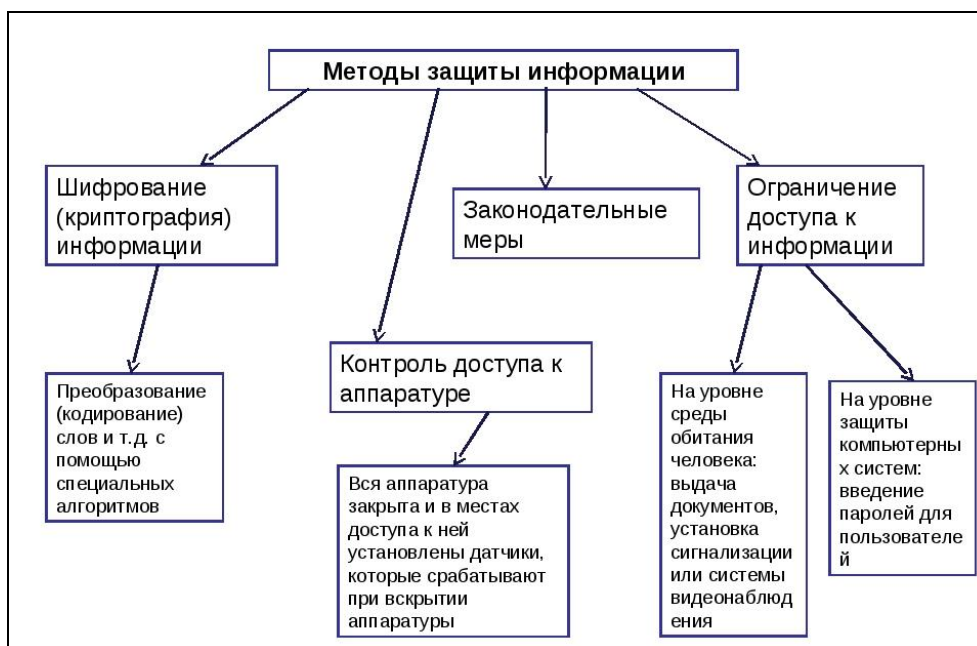


Рисунок 2 – Методы защиты информации

Одним из оптимальных и эффективных методов обеспечения защиты информации, считается шифрование информации. Суть обеспечения защищенности информации путем использования шифрования (криптографии) заключается в том, что для предотвращения несанкционированного доступа к какому-либо сообщению оно зашифровывается. Когда санкционированный пользователь получает зашифрованное сообщение, он его дешифрует или раскрывает его путем использования обратного преобразования программы. Данные методы шифрования основываются на использовании специального алгоритма, который запускается шифрующим ключом, который представляет собою уникальное число или набор чисел.

Большинство организаций, использующие метод шифрования в качестве метода защиты безопасности их информационной системы, считают, что используя один алгоритм для шифрования информации, их конфиденциальности ничто не угрожает. Однако, зачастую это далеко не так и одного алгоритма не хватает, чтобы скрыть всю информацию, тем самым риск потери конфиденциальной информации увеличивается. Одним из решений для минимизации риска, предлагается повысить защищенность конфиденциальной информации путем комбинации методов жаргонов.

Методы жаргонов – это методы шифрования, суть которых заключается в том, что символы исходного текста, заменяются одним или несколькими символами, ранее созданным алфавита в соответствии с принятым правилом преобразования. Каждый из методов имеет достаточную защиту скрываемой информации от дешифровки и легкости определения наличия скрываемой информации. На рисунке 3 проиллюстрирован авторский метод шифрования, основанный на комбинации других методов жаргонов.

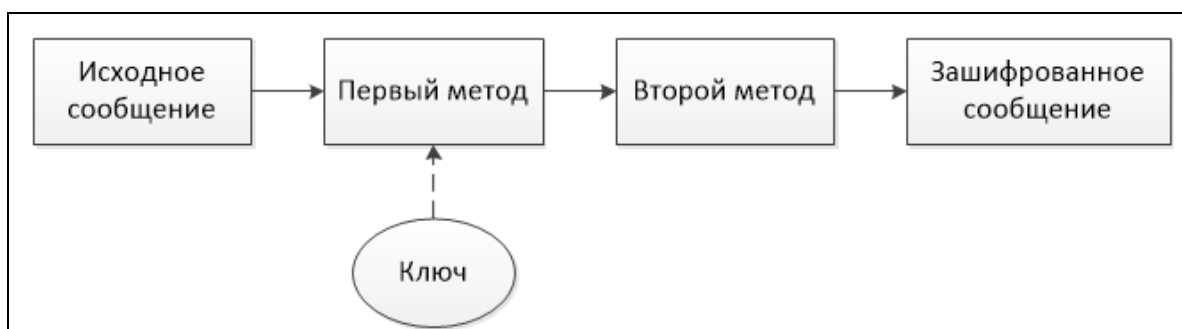


Рисунок 3 – Уникальный метод шифрования

Первым задействованным методом является многоалфавитная подстановка. В данном методе для замены символов исходного текста используются несколько алфавитов. Обычно алфавиты для замены образованы из символов исходного алфавита, записанных в другом порядке. В данном методе для шифрования используется таблица, которая представляет из себя квадратную матрицу с размерностью $N \times N$, где N – это количество символов в алфавите. В первой строке матрицы записываются символы в порядке очередности их в исходном алфавите, а во второй аналогичную последовательность символов, но с циклическим сдвигом влево на одну позицию, в третьей – сдвиг увеличивается на две позиции и т.д. Далее для шифрования текста выбирается ключ, который представляет собой некоторый набор символов исходного алфавита. Затем, из полной матрицы выписывается подматрица шифрования, включающая первую строку и строки матрицы, начальными символами которых являются последовательно символы ключа. На рисунке 4 были составлены матрицы шифрования, где в качестве ключа было выбрано слово «весна».

АБВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ
ВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯАБ
ЕЖЗИКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯАБВГД
НОПРСТУФХЦЧШЩЪЫЬЭЮЯАБВГДЕЖЗИКЛМ
СТУФХЦЧШЩЪЫЬЭЮЯАБВГДЕЖЗИКЛМНОПР

Рисунок 4 – Подматрица шифрования

В процессе шифрования (рисунок 5) под каждым символом шифруемого текста записывают символ ключа, повторяющие ключ требуемое число раз, затем шифруется текст по таблице шифрования, заменяется символами, расположенными на пересечениях линий, соединяющих символы текста первой строки таблицы и символы ключа, находящейся под ней.

ИСХОДНЫЙ ТЕКСТ – МЕТОД ПЕРЕСТАНОВКИ	АБВГДЕЖЗИКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ
КЛЮЧ – ВЕСНА ВЕСНАВЕСНАВЕ	ВГДЕЖЗИКЛМНО ПРСТУФХЦЧШЩЪЫЬЭЮЯАБ
ЗАШИФРОВ.ТЕКСТ – ОЛВЬД СЛАТСФЕЭЪВМО	ЕЖЗИКЛМ НОПРСТУФХЦЧШЩЪЫЬЭЮЯАБВГД
	НОПРСТУФХЦЧШЩЪЫЬЭЮЯАБВГДЕЖЗИКЛМ
	СТУФХЦЧШЩЪЫЬЭЮЯАБВГДЕЖЗИКЛМНОПР

Рисунок 5 – Механизм шифрования многоалфавитной заменой

Далее, по окончании первого метода, зашифрованный текст переходит во второй метод. Вторым методом является пропорциональное или монофиническое шифрование. Это метод замены, в котором уравнивается частота появления зашифрованных знаков для защиты от раскрытия с помощью частичного анализа. Для символов, которые встречаются чаще всего, используется наибольшее количество возможных эквивалентов. Для менее используемых исходных символов может оказаться достаточным одного или двух эквивалентов. При шифровании замена для символа открытого текста выбирается либо случайным, либо определенным образом (например, по порядку). При использовании данного метода, в качестве замены символов можно выбрать числа, как в таблице 1.

Таблица 1 – Таблица с вариантами замены символов

Символ	Варианты замены				
	А	760	128	351	201
Б	101				
В	210	106			
Г	351				
Д	129				
...					
Ю	570				
Я	216	104			

пробел	751	769	758	801	849
--------	-----	-----	-----	-----	-----

По окончанию второго метода, сам процесс шифрования текста полностью заканчивается. Таким образом, проходя двухэтапное шифрование, окончательный вид текст принимает вид, который будет сложен для определения, сторонним лицом, наличия скрываемой информации, и сложен для дешифровки.

Используя предложенный уникальный метод шифрования, организация повышает уровень информационной безопасности на более серьезный уровень, чем было изначально.

Выводы

Конфиденциальная информация является важнейшей составляющей любых информационных отношений. Основной целью защиты конфиденциальной информации состоит в том, чтобы предотвратить ее утечку и овладение этой информацией сторонними лицами.

Не менее важной проблемой защиты конфиденциальной информации, является состояние информационной безопасности, которая обеспечивает защиту информации.

Для повышения защиты конфиденциальности информации был предложен авторский метод, который базируется на использовании нескольких методов жаргонов. Используя данный метод в качестве основного алгоритма для шифрования информации, организация повышает уровень информационной безопасности.

Литература

1. Академик [Электронный ресурс]. – URL: <https://dic.academic.ru/dic.nsf/ruwiki>
2. Библиофонд. Электронная библиотека студента [Электронный ресурс.] – URL: <https://www.bibliofond.ru/view.aspx?id=657009>
3. Дошина А. Д., Михайлова А. Е., Карлова В. В. Криптография. Основные методы и проблемы. Современные тенденции криптографии [Текст] // Современные тенденции технических наук: материалы IV Междунар. науч. конф. (г. Казань, октябрь 2015 г.). — Казань: Бук, 2015. — С. 10-13.
4. Крысин А. В. Информационная безопасность. Практическое руководство — М.: СПАРК, К.:ВЕК+,2003
5. Данилов А.Д. Ценность информации // Технологии информационного общества. 2011. № 10. С. 137-140.
6. Дутов В. Предотвращение утечек информации // Управление рисками организации. 2010. № 3. С. 3-5.

УДК 003.628

ВЛИЯНИЕ SMM-ДИЗАЙНА НА ПРОДВИЖЕНИЕ И ВОСПРИЯТИЕ БРЕНДА

Шепеленко Н.С., Руденко М.П.

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: nikitashepelenko@mail.ru

Аннотация:

Шепеленко Н.С., Руденко М.П. Влияние SMM-дизайна на продвижение и восприятие бренда. Рассмотрены визуальные эффекты в маркетинге социальных сетей, проведен анализ правил по созданию эффективного визуального контента.

Annotation:

Shepelenko N.S., Rudenko M.P. The influence of SMM design on brand promotion and perception. Visual effects in marketing of social networks are considered, the analysis of rules on creation of effective visual content is carried out.

Ключевые слова: дизайн, smm, реклама, социальные сети, Instagram, Facebook

Актуальность исследования. Быстрый темп развития социальных сетей привел к тому, что теперь они не только платформа для общения, как это было задумано изначально, а также эффективная интернет-площадка для рекламы. Большинство брендов обращаются в крупные диджитал-агентства, для того, чтобы найти новых потребителей в интернете. Следовательно, актуальной задачей является создание эффективного визуального контента для социальных сетей с целью привлечения аудитории.

Цель исследования. Проанализировать существующие правила по созданию эффективного визуального контента в социальных сетях.

Основная часть. Люди окружены информацией. Она приходит отовсюду, будь то онлайн или оффлайн. Наша сенсорная система получает сигналы и передает их в мозг. Процесс распознавания и интерпретации входных данных называется восприятием. Среди всех видов восприятия (звук, осязание, вкус, запах и т.д.), визуальное восприятие является первичным, так как 90% информации, поступающей в наш мозг, является визуальной. Исходя из этого, как маркетологи, так и дизайнеры признают важность использования визуальных материалов в маркетинговых стратегиях. Однако, большинство опытных дизайнеров не всегда создают эффективный и продающий визуальный контент.

Визуальные эффекты в маркетинге социальных сетей. Когда дело доходит до маркетинга в социальных сетях (SMM), графический дизайн является решающей частью общей стратегии контента. Независимо от того, насколько велик текст, никто не заметит его, если он не сопровождается хорошими визуальными эффектами, чтобы привлечь внимание людей в середине просмотра. Рассказывая историю бренда с помощью графики, важно иметь в виду стратегические цели, применимые к конкретному маркетинговому ходу [1]. В зависимости от типа бизнеса, SMM может работать для различных целей. Поэтому дизайнеры должны разрабатывать визуальные материалы на основе бизнес-модели и маркетинговой стратегии. Не только маркетинговые цели, но и социальные сети влияют на визуальный дизайн. Если компания общается со своей аудиторией Facebook, Twitter и Вконтакте официальным, а не дружественным образом, дизайн должен поддерживать тон с подходящими цветами и типографикой. В то время как визуальные материалы для Instagram могут быть более непринужденными и расслабленными (рис. 1).

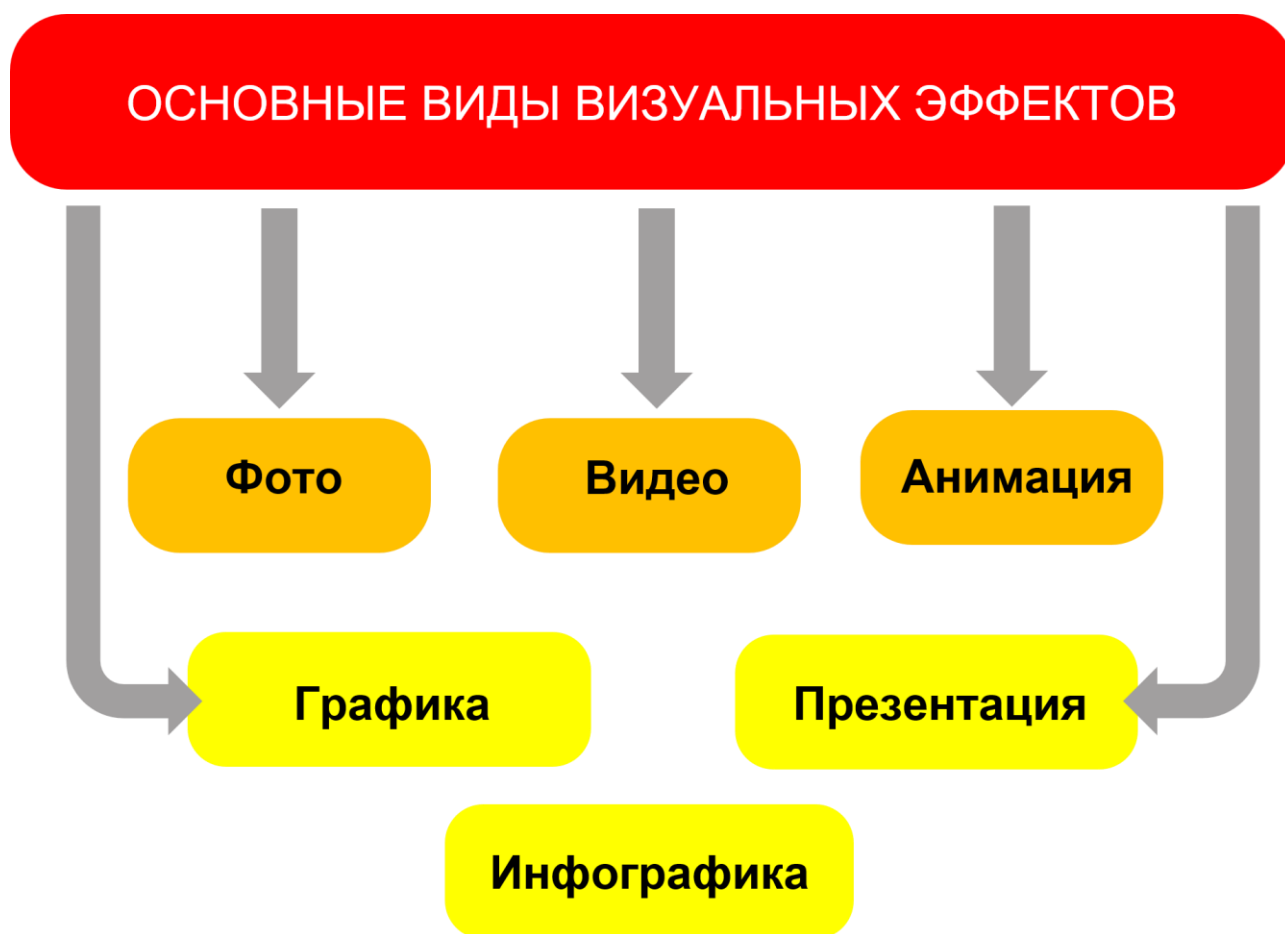
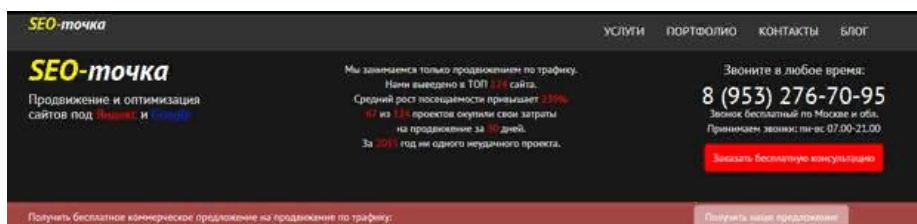


Рис. 1 — Схема основных визуальных эффектов

Правила создания эффективного визуального контента.

Путем многих исследований маркетологи смогли выяснить, какой дизайн является эффективным, хорошо воспринимается аудиторией и привлекает новых клиентов (рис. 2,3). Благодаря этому, сформировались правила, которые помогут любому дизайнеру не ошибиться во время создания визуального контента [3].



Мы берем на продвижение по трафику лишь 10% обратившихся к нам клиентов!



Оптимизация сайта и продвижение по трафику:

1. Увеличение кол-ва целевых посетителей вашего сайта
2. Увеличение продаж за счет роста трафика
3. Оптимальное соотношение цены/качества заявки
4. Снижение показателя отказа
5. Высокая окупаемость и стабильный результат

Оставьте заявку сейчас!

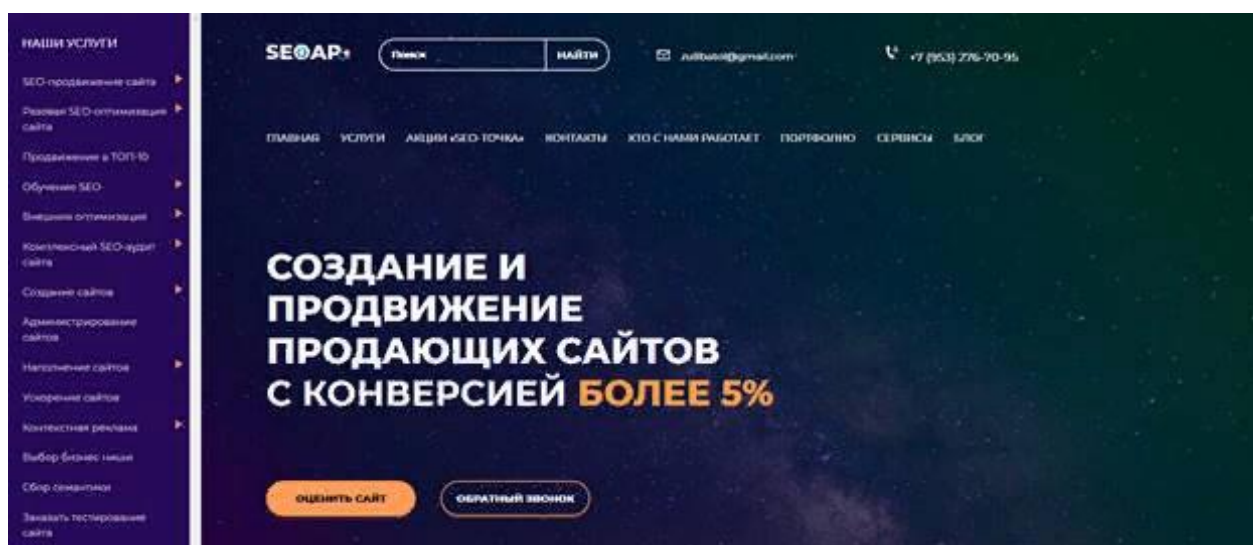
Введите свой e-mail, и через 4 часа с вами свяжется специалист по продвижению!

ваш адрес электронной почты

Оставить заявку сейчас!
Это бесплатно

Заполнение заявки Вас ни к чему не обязывает!
Вы можете отозваться в любой момент!

а)



б)

Рис. 2 — Пример редизайна сайта: а) сайт до редизайна; б) сайт после редизайна

Топ	Яндекс	Google
Топ 3	8%	8%
Топ 5	13%	8%
Топ 10	26%	8%
Топ 20	69%	8%
Топ 50	91%	26%
Топ 100	91%	56%

ДО редизайна

Топ	Яндекс	Google
Топ 3	8%	8%
Топ 5	21% +4%	8%
Топ 10	47%	8%
Топ 20	95%	8%
Топ 50	100%	39% -4%
Топ 100	100%	73%

После редизайна

Рис. 3 — Посещаемость сайта: до и после

Правила по созданию эффективного визуального контента в социальных сетях

1. *Цвет.* Цвета обладают необычайной способностью влиять на настроение людей и вызывать различные эмоции в зависимости от их культурного и личного фона. Выбирая правильный тон, дизайнер должен иметь в виду посыл компании и специфику конкретной социальной сети.

Основной задачей является сделать визуальный контент привлекательным. Немногие пользователи заходят на страницу компании, чтобы проверить последние публикации. Большинство подписчиков страницы просматривают обновления, прокручивая Ленты Новостей. Таким образом, у дизайнера есть меньше секунды, чтобы заставить пользователя остановиться и прочитать сообщение компании.

Дизайнеру в этом случае следует отклоняться от однообразных цветовых сочетаний. Попробовать использовать новые оттенки, даже если у бренда уже есть собственная цветовая палитра, которая отражает его индивидуальность. Также следует использовать яркие цвета, чтобы повысить вовлеченность в рекламный посыл.

2. **Контраст.** Человеческие глаза любят контраст. Контраст может быть связан с цветом, размером, текстурой, формой, макетом, типом и т. д. Это провоцирует зрительные ощущения человека и привлекает внимание, что очень важно в наши дни, когда пользователи листают Ленту Новостей очень быстро (рис. 4).

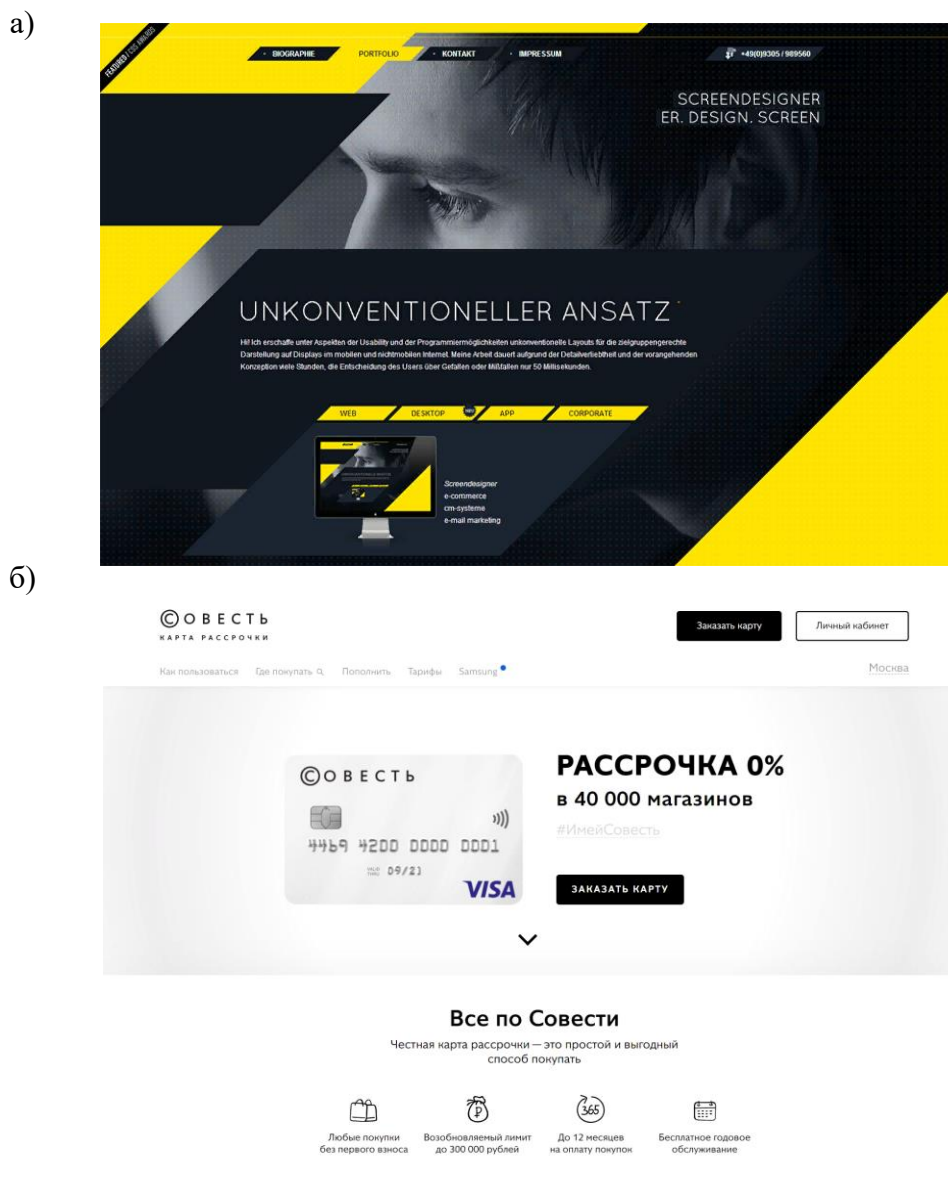


Рис. 4 — Пример сайта с использованием контраста (а) и без контраста (б)

Контраст - это полезный инструмент для организации нескольких графических объектов и фокусировки на определенном визуальном элементе. Контраст может усилить идею дизайна, соединить и дополнить элементы, или сделать все изображение бессмысленным и запутанным.

3. *Типографика.* Заставить людей прочитать текст на картинке еще сложнее, чем заставить их заметить картинку. Вот почему выбор правильного шрифта и структуры имеет важное значение при разработке визуальных элементов для социальных сетей.

По данным портала Statista, к 2020-му году число пользователей соцсетей со смартфонами достигнет 186,3 млн в месяц. Поэтому следует убедиться, что рекламное сообщение будет хорошо читаться с экрана смартфона [2].

Любям не нравится сложная типографика в социальных сетях. Пользователи хотят, чтобы сообщение было ясным и понятным, поэтому следует сделать читабельность приоритетом. Однако сильно сфокусированные на тексте изображения могут быть скучными. Чтобы избежать этого, нужно попробовать поиграть с начертаниями, размером шрифтов.

4. *Стиль.* Стиль в дизайне имеет широкое определение. Это сложное сочетание цветов, шрифтов, композиций и так далее. Стиль делает дизайн узнаваемым.

Некоторые дизайнеры считают, что придерживаться одного и того же стиля не выгодно для их портфолио. Для профессионального роста важно опробовать различные методики и подходы. Это применимо к дизайну изображений в социальных сетях. Следовательно, нужно разнообразить стилистику бренда, чтобы удивить аудиторию оригинальными дизайнерскими решениями, и они никогда не устанут от контента бренда.



Комбо



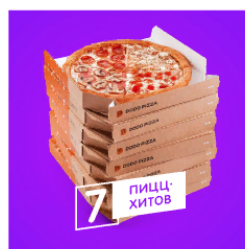
2+2

Понравится всем! Хит продаж – Пепперони 30 см, нежная Маргарита 30 см и две бутылочки Соса-Cola 0,5 л – Классическая и Зеро.



Пепперозавр

Утолит даже зверский аппетит! Пять сытных пицц 30 см с пепперони: Мясная, Пепперони, Двойная пепперони, Итальянская и Четыре сезона.



Большая 7-ка

Счастливый набор из семи пицц 30 см для большой компании: Пепперони, Сырный цыпленок, Цыпленок барбекю, Ветчина и грибы, Маргарита, Четыре



Две Соса-Cola по суперцене

Две бутылочки Соса-Cola 0,5 л по суперцене: Классическая и Зеро



Рис. 5 — Использование стиля бренда на сайте и в рекламных постах

5. *Шаблоны.* Готовые шаблоны визуализации рекламных постов будут не только существенно экономить время дизайнеру (т.к. любой маркетолог сможет самостоятельно отредактировать контент), но и поможет повысить узнаваемость бренда.

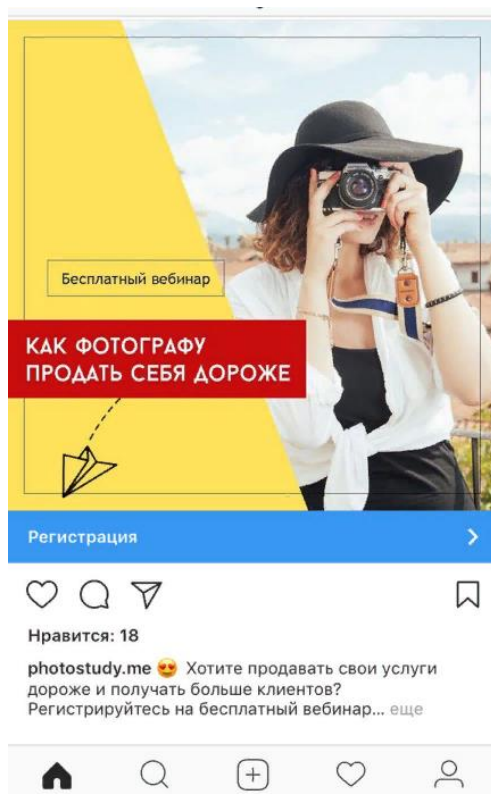
6. *Брендинг.* Использование фирменного стиля в рекламной графике существенно увеличит узнаваемость бренда и поможет аудитории изучить продукцию компании. Однако, брендинг рекламного контента не должен отвлекать пользователя от самого посыла сообщения (рис. 5).

7. *Иерархия.* Дизайнер может помочь пользователю быстро и точно понять информацию, благодаря визуальной иерархии (основанной на Теории Гештальта). Согласно этой теории, люди склонны воспринимать целое до того, как заметят отдельные объекты.[1]

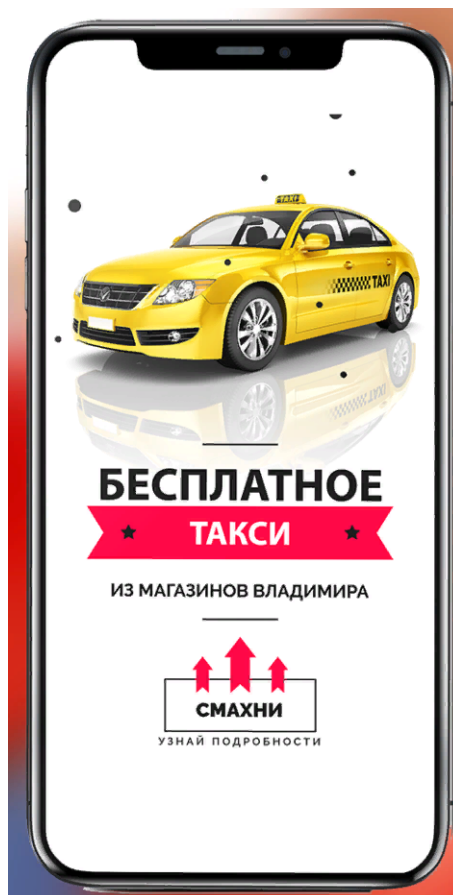
Через составленные визуальные характеристики (размер, цвет, контраст, повторение, близость, текстура, выравнивание, etc.) дизайнеры могут заставить глаза пользователей следовать определенному пути чтения и привлечь внимание пользователей к наиболее важным элементам.

8. *Призыв к действию.* В социальных сетях недостаточно сделать привлекающий внимание дизайн. Маркетологи хотят, чтобы связь с аудиторией была двусторонней, чтобы люди взаимодействовали с брендом. Поэтому поощрительный текст и элементы, призывающие пользователя к действию помогут дизайнеру сделать визуальный контент более эффективным (рис. 6).

9. *Четкость и ясность.* Любой дизайн (простой или сложный), должен быть понятен аудитории. Дизайнер должен держать баланс между простотой и информативностью. Посыл бренда должен быть очевиден, т.к. пользователи очень быстро переводят свое внимание на другие посты в Ленте.



а)



б)

Рис. 6 — Баннеры с призывом к действию (а, б)

Выводы

Рассмотрены визуальные эффекты в маркетинге социальных сетей, проведен анализ правил по созданию эффективного визуального контента. Сделан вывод о том, что визуальная графика является неотъемлемой частью любого маркетингового плана социальных медиа, и вклад дизайнера в успех маркетинговой стратегии неоспорим. Дизайнер должен учитывать маркетинговые цели и тон компании при создании любых визуальных эффектов для социальных сетей.

Литература

1. Вики чтение [Электронный ресурс] // Гештальт: искусство контакта [Новый оптимистический подход к человеческим отношениям]. URL: <https://psy.wikireading.ru/23722>
2. Statika [Электронный ресурс] // Number of smartphone social network users from 2014 to 2020 (in millions). URL: <https://www.statista.com/statistics/238636/number-of-mobile-social-network-users-in-the-us/>
3. blog.intage [Электронный ресурс] // Эволюция дизайна в SMM: от истоков к трендам 2019. URL: <https://blog.ingate.ru/detail/evolyutsiya-dizayna-v-smm-ot-istokov-k-trendam-2019/>
4. Genius Marketing [Электронный ресурс] // 10 законов SMM-маркетинга: чего от вас ждут клиенты. URL: <https://geniusmarketing.me/lab/10-zakonov-smm-marketinga-chego-ot-vas-zhdut-klienty/>

УДК 004.588

РАЗРАБОТКА ИГРЫ «ЗАНИМАТЕЛЬНАЯ МАТЕМАТИКА» СРЕДСТВАМИ CONSTRUCT 2

Беспалова Е.С., Киселева О.В.

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: katytyan@gmail.com

Аннотация:

Беспалова Е.С., Киселева О.В. Разработка уровня игры «Занимательная математика» средствами Construct 2. Разработан начальный интерфейс уровня игры. Созданы спрайты главного и отрицательных персонажей. Реализованы движения персонажей на сцене. Объясняется принцип сбора объектов.

Annotation:

Bespalova E.S., Kiseleva O.V Development of the game level "Entertaining mathematics" by means of Construct 2. An initial level one interface has been developed. Sprites of the main and negative characters are created. Implemented the movement of characters on stage. The principle of collecting objects is explained.

Общая постановка проблемы

Визуально привлекательный и удобный пользовательский интерфейс - ключевой показатель качества приложения. В сочетании с грамотной структурой и логичной навигацией по разделам проекта, он привлекает посетителей и улучшает функциональность самого приложения.

Главная задача в такой работе как проектирование игровых интерфейсов - максимально упростить жизнь пользователю, сделать так, чтобы он достигал желаемый результат, затрачивая минимум усилий. Как раз для этого служат кнопки, вкладки и элементы оформления. Удобная форма входа, выбора уровня, общепринятые значки, например, схематическое изображение ноты в качестве кнопки “звук”. Это понятно для каждого и не нуждается в объяснении.

Создание платформы и фона

На сцену добавляются подготовленные объекты для параллакса фона с помощью переноса в окно приложения и располагаются в начале слоя под названием “Background”. Создается отдельный слой “Ground”. Для добавления платформы двойным левым кликом мыши открывается окно Insert New Object и выбирается Tiled Background. В всплывающем окне выбирается “открыть имеющиеся изображение”. Необходимо включить сетку для привязки объектов с лучшим расположением. Все остальные блоки делаются по аналогии (Рис. 1).

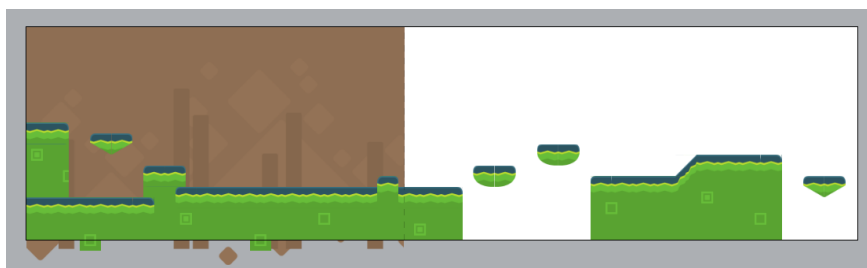


Рис. 1. Создание фона и платформы

Для платформ, на которых будет ходить персонаж, выставляется поведение Behavior – Solid. Платформы, на которые можно запрыгнуть снизу выставляется Behavior – Jump thru.

Чтоб присутствовал эффект параллакса в окне Layer properties в значении Parallax вписывается 0, 0.

Создание главного персонажа

Создается новый слой под названием “Game”. Для анимации движений главного персонажа были разработаны спрайты, представленные на рис. 2.

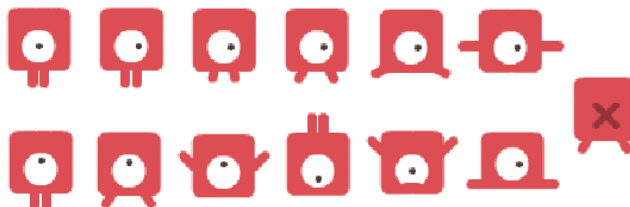


Рис. 2. Спрайты главного персонажа.

На сцену переносится первый спрайт персонажа и выставляется поведение Behavior – Platform. Для анимации движения правым кликом мыши выбирается Edit animations. В сплывающем окне Animations добавляется состояние Walk. В окне Animations frames правым кликом мыши добавляются новые спрайты. Окно редактирования и добавление анимации представлено на рис. 3.

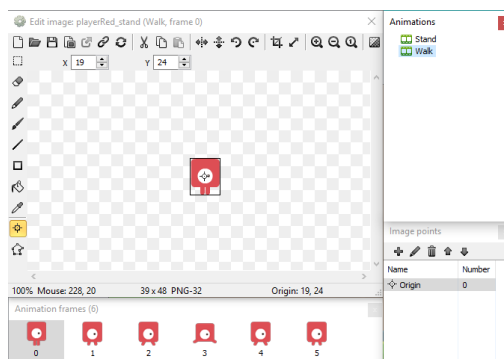


Рис. 3. Окно добавления спрайтов.

Чтоб использовать состояние Walk в файле Event sheet создается условие Add event – Выбираем персонажа – On moved. Затем после добавления блока условия создается действие Add action – Персонаж – Set animation – Вписывается “Walk”.

Для того, чтоб анимация движения остановилась, если персонаж стоит добавляется еще одно условие Add event – Персонаж – On stopped. Затем после добавления блока условия создается действие Add action – Персонаж – Set animation – Вписывается “Stand”. Анимации прыжка и падения сделаны по аналогии.

Создается условие для клавиатуры, чтоб персонаж менял направление взгляда в сторону своего движения. Для этого добавляется условие Add event – Keyboard – On Left arrow pressed – Add action – Персонаж – Set animation – State Mirrored. А также Add event – Keyboard – On Right arrow pressed – Add action – Персонаж – Set animation – State Not Mirrored. Все события, созданные для Главного персонажа представлены на рис. 4.

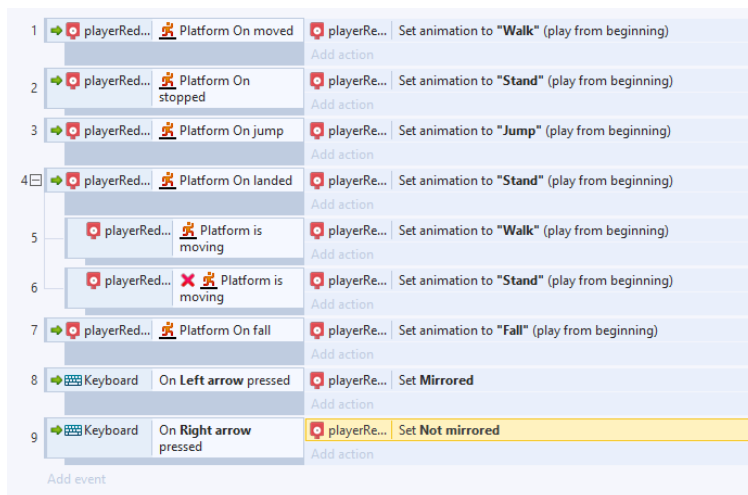


Рис. 4. Условия Главного персонажа.

Создание отрицательных персонажей

Для анимации движений отрицательных персонажей были разработаны спрайты, представленные на рис. 5.

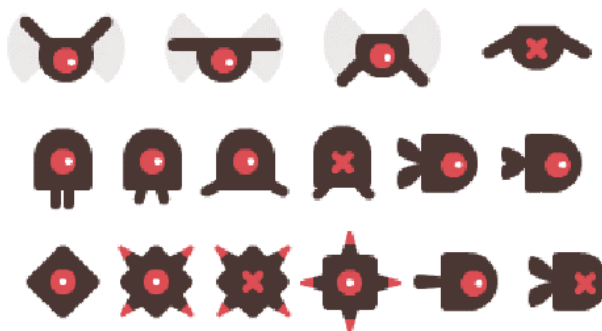


Рис. 5. Спрайты отрицательных персонажей.

На сцену переносится первый спрайт отрицательного персонажа. Необходимо добавить взаимодействие отрицательного персонажа с главным персонажем. У главного персонажа выставляется поведение Behavior – Flash. Создается условие Add event – Персонаж – On collision with another object, затем добавляется действие Add action – Персонаж – Flash – Выставляется параметры на 0.2, 0.2, 1.0. Теперь при столкновении главного персонажа с отрицательным, первый будет мигать.

Для летающего отрицательного персонажа выставляется поведение Behavior – Sine, и параметры Magnitude – 100. Добавляется еще одно поведение Behavior – Sine с параметрами Movement – Vertical, Period – 2, Magnitude – 30. Таким образом теперь этот персонаж вырисовывает горизонтальную восьмерку в воздухе. Создаются аналогичное условие для мигания при столкновении.

Для ходящего отрицательного персонажа выставляется поведение Behavior – Platform с параметром Default controls – No. Также добавляется Instance variables с Name – move, Type – text с начальным значением – right. Далее создается условие Add event – Ходящий отрицательный персонаж – Is on screen, затем добавляется подусловие Add event – Ходящий отрицательный персонаж – Compare instance variable – в поле value вводится "right". Затем создается действие Add action – Ходящий отрицательный персонаж – Simulate control – Right. Создается еще одно подусловие Add event – Ходящий отрицательный персонаж – Is by wall – right. Создается к нему действие Add action – Ходящий отрицательный персонаж – Set value –

в поле value вводится "left". Все проделанные действия создаются и для столкновения с другой стенкой. Создается аналогичное условие для мигания при столкновении. Добавляется действие к 16 подусловию Add action – Ходящий отрицательный персонаж – Set mirrored – Mirrored. Аналогично проделывается для 18 условия с противоположным состоянием Add action – Ходящий отрицательный персонаж – Set mirrored – Not Mirrored.

Для второго летающего отрицательного персонажа выставляется поведение Behavior – Bulet с параметрами Set angle – No, Speed – -300 и начальным состоянием – Disabled. Создается условие Add event – Второй летающий отрицательный персонаж – Is on screen, затем добавляется подусловие Add event – System – Trigger once while true и создается к нему действие Add action – Второй летающий отрицательный персонаж – Set enabled – Enabled. Все события, созданные для всех отрицательных персонажей предоставлены на рис. 6.

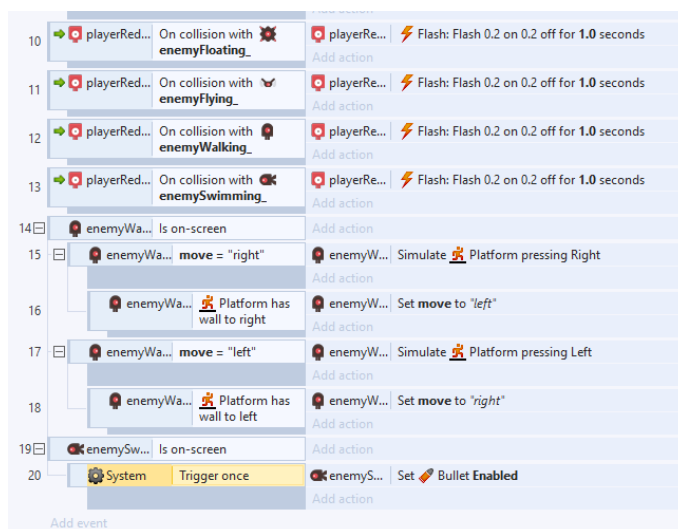


Рис. 6. Условия отрицательных персонажей.

Принцип сбора объектов

Сбор осуществляется, когда главный персонаж, управляемый пользователем, наткнется на объект, удовлетворяющий условия уровня. Если объект оказывается правильным, то он добавляется в собранные правильные объекты. Сбор трех правильных объектов является условием перехода на следующий уровень. Если объект оказывается неправильным, персонаж теряет жизнь. Потеря трех жизней приводит к прохождению уровня снова.

Выводы

Обучающие игры создают идеальные условия для того, чтобы человек вошел в наиболее приятное и креативное состояние — именно поэтому они так нравятся детям.

Игра, как особый вид деятельности, представляет собой процесс, суть которого можно обозначить как отображение и преобразование действительности, в ходе которого происходит усвоение общественного опыта, отработка умений и становление навыков. Компьютерная игра - это компьютерная программа, направлена на организацию этого процесса.

Важное значение имеет то, что развивающие и обучающие компьютерные игры не только эмоционально привлекательны для детей, но и содержательны, а значит их можно подчинить педагогическим целям. Правильное их применение позволяет скорректировать развитие ребенка, сделать процесс усвоения знаний более успешным.

Для реализации уровня игры были использованы программы Adobe Illustrator CC, NWJS, Construct 2.

Литература

1. Bura John, Construct 2 Game Development by Example [Текст] / Bura John // 2014. – С. 59.
2. Lee Stemkoski, Evan Leider, Game Development with Construct 2 [Текст] / 2. Lee Stemkoski, Evan Leider // 2014. – С. 105.
3. Construct 2 [Electronic resource] / Интернет-ресурс. - Режим доступа: www/ URL: <https://www.scirra.com/construct2>
4. Add Behavior dialog [Electronic resource] / Интернет-ресурс. - Режим доступа: www/ URL: <https://www.scirra.com/manual/49/add-behavior>

УДК 004.588

РАЗРАБОТКА СЦЕН ИГРЫ "ЗАНИМАТЕЛЬНАЯ ГРАММАТИКА" СРЕДСТВАМИ UNITY 3D

Демьяненко В.А., Киселева О.В.

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: Lera00708@yandex.ru

Аннотация:

Демьяненко В.А., Киселева О.В. Разработка сцен игры «Занимательная грамматика» средствами Unity 3D. Рассмотрена межплатформенная среда разработки компьютерных игр Unity. Исследованы основы создания игровой сцены и моделей, наложения текстур, работы с освещением. Изучены принципы разработки игровой логики с помощью скриптов.

Annotation:

Demianenko V.A., Kiseleva O.V. Development of scenes of the game "Entertaining grammar" means Unity 3D. The cross-platform development environment of Unity computer games is considered. The basics of creating a game scene and models, applying textures, working with lighting. The principles of game logic development using scripts are studied.

Использование компьютерных игр в образовательном процессе

Применение компьютерных технологий в образовательном процессе – очевидно, учитывая роль компьютера в жизни современного ребенка. Внедрение компьютерных технологий в обучение расширяет границы образовательного процесса: появляются новые методы и формы обучения. Обновление содержания обучения связано с повышением роли самообразования в процессе школьного обучения, формированием у обучаемых современного представления об окружающем мире с учетом стремительного развития компьютерных технологий. Данный подход к обучению обеспечивает переход от механического усвоения знаний к овладению умением самостоятельно приобретать новые знания и совершенствовать их в дальнейшем. Также способствуют выявлению, сохранению и развитию индивидуальных качеств обучаемых.

Компьютерные обучающие игры являются эффективным инструментом современного образовательного процесса. Изучение материала в игровой форме имеет множество достоинств:

1. Повышенная внимательность и увлеченность в процессе обучения.
2. Настойчивость и терпение, не свойственные обычно детям.
3. Мотивация к деятельности детей за компьютером:
 - а) интерес к нестандартному способу обучения;
 - б) исследовательский мотив (желание найти ответ на вопросы);
 - в) мотив успешного решения познавательных задач.
4. За компьютером дети предпочитают играть вместе, находя общее решение.

Взрослые только помогают детям советами. Совместные компьютерные игры позволяют снять ряд трудностей в общении детей.

Unity3D, как средство разработки компьютерной игры

Unity — это мультиплатформенный инструмент для разработки двух- и трёхмерных приложений и игр, работающий под операционными системами Windows, OS X, Android, Apple iOS, Linux.

Для разработки, обучающей 3D игры была выбрана среда разработки Unity, так как она специально адаптирована именно на качество графики, в том числе с поддержкой 3D. Unity3d идеально подходит для разработки 3D игры.

Основной концепцией Unity3d является использование в сцене легко управляемых объектов, которые, в свою очередь, состоят из множества компонентов. Создание отдельных игровых объектов и последующее расширение их функциональности с помощью добавления различных компонентов позволяет бесконечно совершенствовать и усложнять проект.

Unity — это полноценный игровой движок, рассчитанный на то, что весь процесс разработки происходит в поставляемом в комплекте Редакторе. В этом редакторе компонованы сцены будущей игры, связывая игровые ресурсы и код в интерактивные объекты. Именно он позволяет быстро и рационально создавать профессиональные игры, обеспечивая продуктивность разработчики и предоставляя исчерпывающий перечень самых современных технологий в области видеоигр. Реализованная с помощью этой технологии игра получает все функциональные возможности, кроме того, добавляются ее собственные игровые ресурсы и код игрового сценария. У Unity есть два основных преимущества перед другими передовыми инструментами разработки игр: чрезвычайно производительный визуальный рабочий процесс и мощная межплатформенная поддержка.

Так же Unity предоставляет наиболее широкие возможности для программирования приложений в ОС Windows. Unity позволяет создавать кроссплатформенные 3D приложения, в Unity доступны три языка: JavaScript, C# и разновидность Python's Boo, что предоставляет разработчику выбор наиболее удобного языка для разработки.

Создание основной сцены игры

Главная сцена игры – это красочная лесная поляна, на которой расположено множество домиков. Каждый домик – отдельная мини-игра(уровень). Для главной сцены необходимо создать:

- Ландшафт игровой площадки
- Деревья
- Домики

Для ландшафта игровой площадки были созданы текстуры для покрытия сцены в графическом редакторе Adobe Photoshop:

- Текстура земли - для прорисовки тропинок к домикам;
- Текстура травы– для детализации площадки;
- Текстура травяного покрытия – для покрытия площадки.



Рис. 1. Текстуры

Для сцены был использован игровой объект Terrain, который представляет прямоугольную плоскую поверхность и включает все инструменты необходимые для создания и редактирования ландшафта. С помощью Raise / Lower Height и Smooth Height компонента Terrain была увеличена/уменьшена высота участков ландшафта и сглажены резкие переходы вершин и углублений. По краям сцены сформированы высокие холмы для визуальной ограниченности сцены. Будущий персонаж не сможет перейти холмы и выйти за границы сцены. После создания ландшафта он был текстурирован используя инструмент Paint Texture, в который добавлены созданные ранее текстуры. Было выполнено покрытие ландшафта травой и землей, нарисованы тропинки к будущим домикам. Также, с помощью Paint Details добавлены участки с растущей травой на поверхности земли и холмах, для детализации и реалистичности сцены.

Для освещения сцены и объектов был использован источник света Light. Тип источника – Direction, который оказывает влияние на все объекты на сцене и представляет собой множество параллельных лучей. В свойствах света был изменен Color - цвет источника на теплый желтый (более теплый и подходящей для «солнца»), Intensity - интенсивность освещения. Данный тип имитирует освещение солнечным светом и моделирует соответствующие тени. Он идеально подходит для освещения лесной поляны.

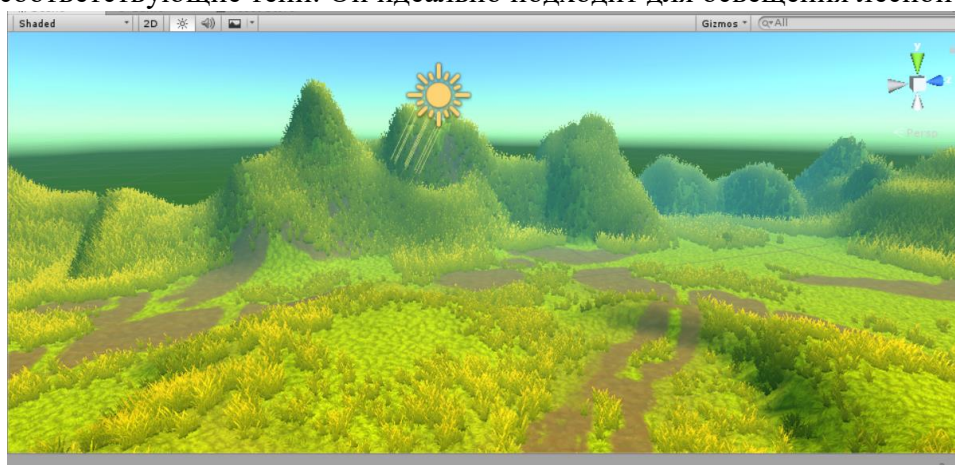


Рис. 2. Ландшафт сцены

За визуализацию окружающей среды вокруг сцены отвечает панорамная текстура Skybox, она остается позади всех элементов сцены и вращается относительно текущей камеры. В каком бы направлении не смотрела камера, она будет отображать небо. Был выбран стандартный Skybox material, который симулирует вид неба в дневное время суток. А также настроен голубой цвет окружающего освещения, для придания реалистичности отображения дневного света.

Для деревьев были созданы текстуры для коры дерева и листвы. Текстуры включены в материалы и выбраны шейдеры из категории Nature, предназначенные для растительности - Tree Soft Occlusion Bark и Tree Soft Occlusion Leaves. Скорректированы их настройки: четкость перехода от материала к фону, базовое освещение и затенение материала.

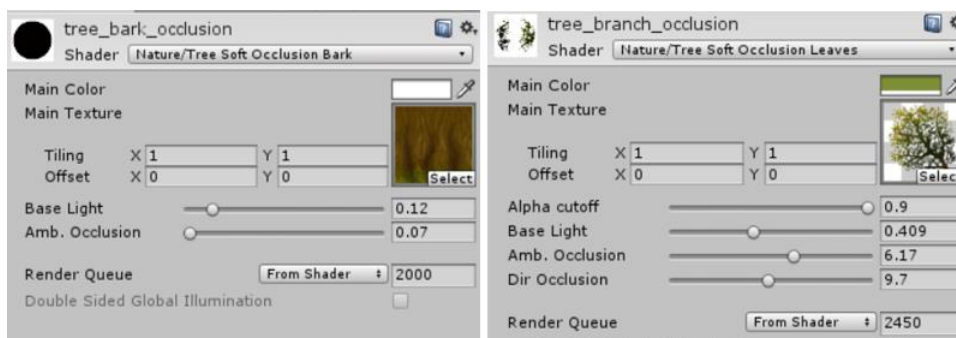


Рис. 3. Материалы для деревьев

Для создания деревьев был использован встроенный модуль Unity3D Tree Editor. Дерево состоит из корня, основной ветви(ствола), дочерних ветвей(веток) и листьев. Были использованы свойства каждого компонента: длина и радиус ствола/веток, кривая изгиба ствола/веток, расположение ветвей, тип распределения листьев, количество и размер листьев. На все составляющие были наложены соответствующие материалы. По аналогии созданы несколько видов деревьев, для того чтобы придать красочности и разнообразия игровой сцене.

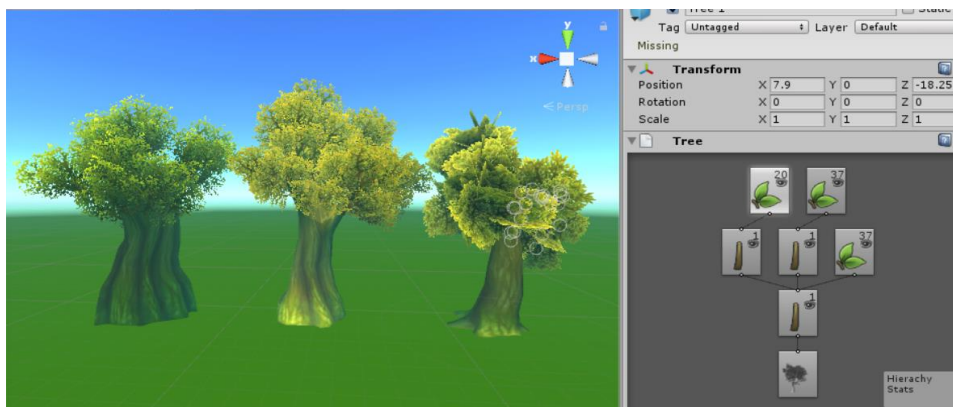


Рис. 4. Деревья

Используя инструмент Paint Trees, который находится в настройках компонента Terrain, были размещены деревья на ландшафте сцены.

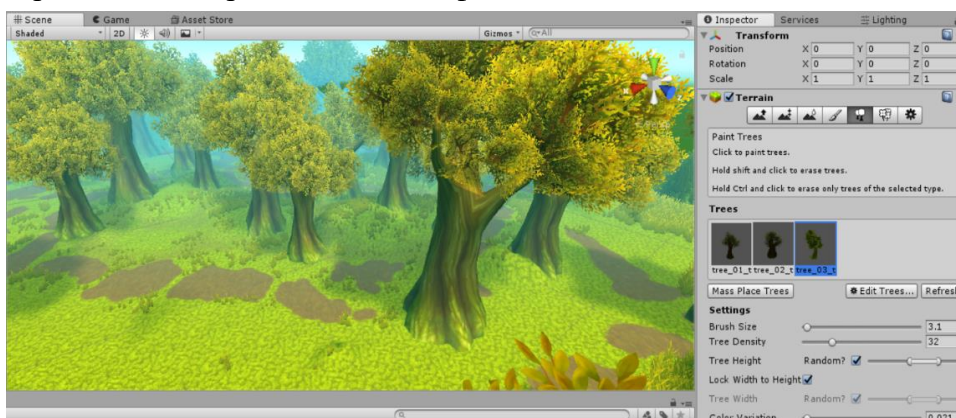


Рис. 5. Размещение деревьев на ландшафте

Для создания 3D модели домиков была использована программа для трёхмерного моделирования Blender. В ходе моделирования использованы стандартные геометрические примитивы Кубы и Цилиндры, а также инструмент Выдавливание вершин, ребер, граней.

Для текстурирования 3D модели была создана UV-развертка в Blender. Двумерная карта объекта, содержащая все элементы модели, закрепленные к определенным местам на текстуре. Выбраны соответствующие грани и ребра (линии разреза) для развертки и скорректировано расположение деталей модели. Развертка сохранена и перенесена для отрисовки текстур в Adobe Photoshop. Карта текстур была нанесена на модель в Blender.

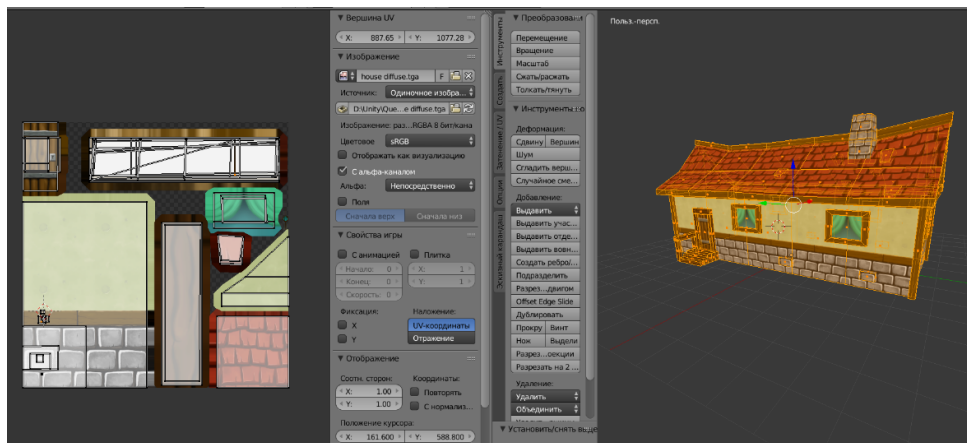


Рис. 6. Создание домика

На готовую смоделированную сцену был помещен игровой персонаж, управление над которым производится от 1го лица. Для этого в Unity реализован стандартный префаб FirstPersonController. Он содержит 2 объекта:

1) First Person Controller - родитель группы, к которому применен стандартный скрипт FirstPersonControl, который предоставляет функциональность движения. К этому объекту также прикреплен Character Collider, который выступает в качестве компонента, который обеспечивает объекту физическое присутствие, т.е. позволяет взаимодействовать с другими объектами, он специально разработан для управления персонажем в мире.

2) Camera - область просмотра, она расположена на уровне глаз и управляется стандартными скриптами, позволяя игроку перемещать весь родительский объект и камеру независимо. Таким образом, игрок может ходить и осматриваться одновременно.

Для предотвращения прохода объектов сквозь друг друга, на домики, деревья и игровую площадку были прикреплены Box Collider, Mesh Collider и Terrain Collider соответственно. Это невидимые сетки, которые имитируют форму объектов и сообщают о наличии столкновений с другими объектами.

Для того, чтобы попасть на игровой уровень, нужно зайти в домик. Данная задача реализуется с помощью скрипта. Для начала возле входа в дом был создан триггер - прозрачный куб с Box Collider, соответствующий размерам двери. Он обрабатывается скриптом загрузки новой сцены при столкновении игрока с кубом. Также создан скрипт с одной переменной, которой мы будем передавать имя следующей сцены для загрузки.

Создание игрового уровня

Игровой уровень – комната внутри домика, заполненная различными соответствующими «правильными» и «неправильными» объектами, которые будут определены в задании. Целью игрока является сбор всех «правильных» объектов, после чего игрок может перейти к следующему уровню.

Для моделирования стен, пола, потолка, окон и двери был использован 3D-объект Cube. А также подготовлены 3D-модели стола, стульев и фруктов для заполнения сцены при помощи программы для трёхмерного моделирования Blender. Созданы текстуры для их покрытия с помощью графического редактора Adobe Photoshop.

На сцену помещен источник света типа Point, настроен его цвет и интенсивность освещения. Это точечный источник света, который имитирует свет лампы и прекрасно подойдет для освещения комнаты.

Каждый «правильный» и «неправильный» объект (стол, стулья, фрукты) имеет табличку с надписью, которая должна быть читабельна с любого ракурса. Она смоделирована с помощью программы Blender, на которую наложены текстуры с использованием UV-развертки.

В верхней части экрана отображено задание мини-игры и прогресс её прохождения при помощи UI элемента Text, который помещен в компонент Canvas. Он представляет собой абстрактное пространство, в котором производится отрисовка и настройка всех элементов UI. Для отображения текста поверх всех объектов сцены и вне зависимости от направления камеры был выбран режим холста Screen Space – Overlay.

На готовую смоделированную сцену был помещен игровой персонаж, управление над которым производится от 1го лица. Для этого в Unity реализован стандартный префаб FirstPersonController.

Для предупреждения прохода персонажа сквозь объекты сцены, каждый объект содержит соответствующий его форме Collider.

Чтобы реализовать сбор «правильных»/ «неправильных» объектов были созданы 2 аналогичных скрипта. На все объекты устанавливаются соответствующие триггеры и добавляется скрипт, применяемый метод OnTriggerEnter. Данный метод срабатывает, когда какой-либо коллайдер попадает в зону триггера (столкновение персонажа с объектом). При столкновении с «правильным» объектом, переменная содержащая количество собранных предметов увеличивается на 1, а объект удаляется. При столкновении с «неправильным» объектом, отображается надпись: «Не верно!».

Отдельный скрипт добавлен игроку. Скрипт выводит текстовую информацию на экран о количестве собранных предметов и завершении уровня.

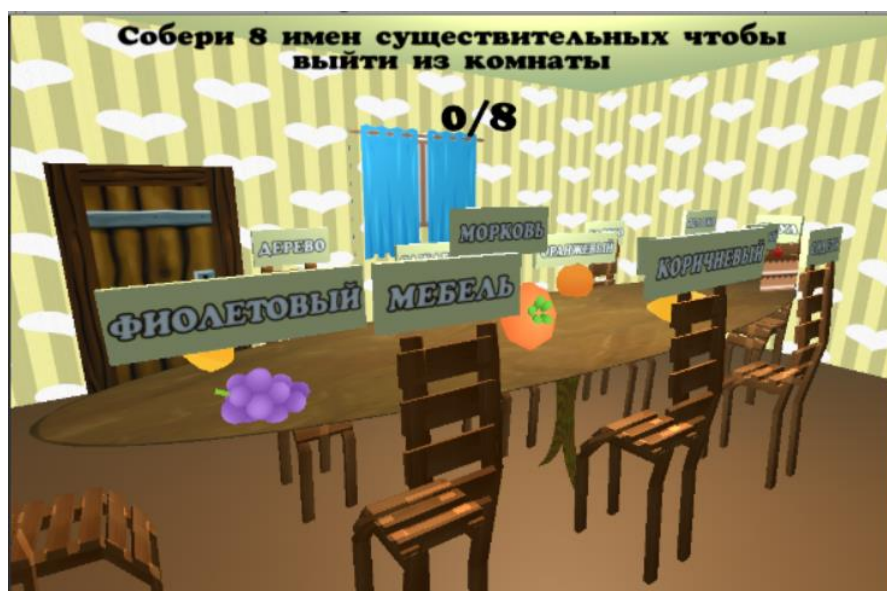


Рисунок 8 – Комната уровня

Выводы

В качестве средства разработки была выбрана Unity 3D. Элементы сцен разработаны с помощью программы для трехмерного моделирования Blender и графического редактора Adobe Photoshop. Была разработана главная сцена и игровой уровень. В ходе разработки созданы и детализированы ландшафт и деревья с использованием встроенных редакторов Unity, смоделированы 3d-модели и наложены текстуры для заполнения сцен игры. Также

были реализованы скрипты для перехода в игровой уровень, сбора предметов и наращивания очков за сбор «правильных» предметов, отображения информации о задании и прогрессе прохождения уровня.

Литература

4. Хокинг Дж. Unity in Action / Unity в действии. Мультиплатформенная разработка на C# [Текст] / Хоккинг Дж.- СПб.: Питер, 2016. – 336с.
5. Unity-Manual: Руководство пользователя по Unity (2017.4) / Интернет-ресурс. - Режим доступа: URL: <https://docs.unity3d.com/Manual/index.html> - Загл. с экрана.
6. Unity Technologies Industries / Интернет-ресурс. - Режим доступа: URL: <http://unity3d.com/unity/industries> - Загл. с экрана.
7. Unity Learn Tutorials / Интернет-ресурс. - Режим доступа: URL: <https://unity3d.com/ru/learn/tutorials> - Загл. с экрана.

УДК 004.92:004.925:004.928

ГЕОМЕТРИЧЕСКОЕ И ДИНАМИЧЕСКОЕ МОДЕЛИРОВАНИЕ СЛОЖНОГО ПРОМЫШЛЕННОГО 3D-ЛАНДШАФТА В СРЕДЕ CINEMA 4D

Верховский А.В., Пчелкин В.Н.

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: fasharonni@gmail.com

Аннотация:

Верховский А.В., Пчелкин В.Н. Геометрическое и динамическое моделирование сложного промышленного 3D-ландшафта в среде Cinema 4D. Рассмотрены возможности среды Cinema 4D для моделирования и построения 3D-ландшафта. Определены преимущества данной среды перед аналогами от других разработчиков. Изучены основные аспекты промышленного ландшафтного дизайна. Смоделирован один из объектов проекта.

Annotation:

Verkhovskiy A.V., Pchelkin V.N. Geometric and dynamic modeling of complex industrial 3D-landscape in the environment of Cinema 4D. The possibilities of the Cinema 4D environment for modeling and building a 3D landscape are considered. The advantages of this environment over analogues from other developers are defined. Studied the main aspects of industrial landscape design. Modeled one of the project objects.

Общая постановка проблемы

В настоящее время для моделирования ландшафта используются достаточно старые и сложные в освоении программы. Среда моделирования Cinema 4D проста в освоении и позволяет дизайнеру самостоятельно моделировать ландшафт, здания, растения, различные объекты и динамику сцены без потребности использования сторонних библиотек, что даёт возможность использовать её для создания иллюстраций, моушн графики, спецэффектов и в во многих других сферах.

Моделирование

Для создания сцены сперва требовалось создать черновой макет расположения зданий

и дорог, который разделён на 4 зоны.[1] В данной работе будет смоделирована и анимирована сцена со складом.

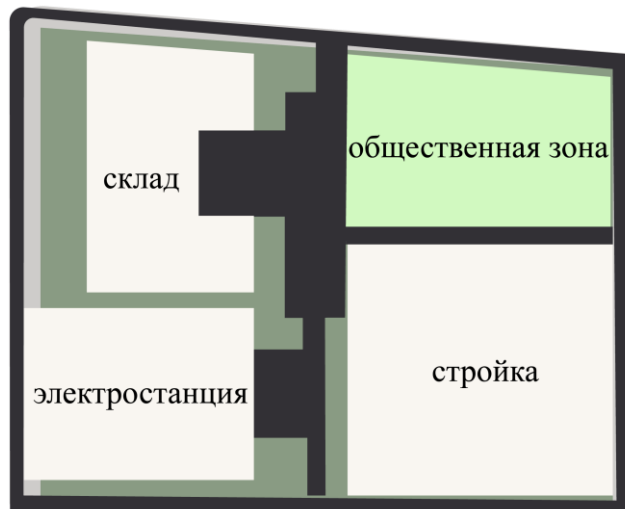


Рис. 1. Макет расположения зданий и дорог

Далее пошёл процесс моделирования зданий.[2] Изначально была создана грубая модель сцены включающая в себя: здание склада и его ограждение. Затем добавлены дополнительные детали на здание, ёмкости для хранения газа, другие мелкие объекты сцены, а также были нанесены текстуры. На рисунке 2 изображено здание и окружающая территория склада в процессе моделирования.



Рис. 2. Модель склада и окружающей территории в процессе моделирования

Для создания динамики в сцене был смоделирован и импортирован погрузчик, а также сделана анимация открывания и закрывания гаражных дверей. Движения погрузчика задано нарисованной линией и синхронизировано с дверьми.

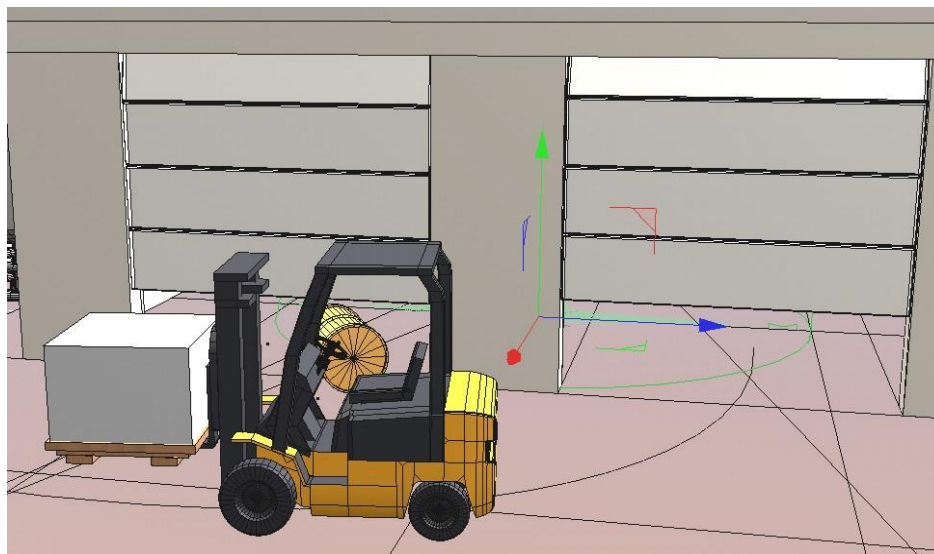


Рис. 2. Модель погрузчика и анимация двери

Под конец в сцену со складом был добавлен свет для просчитывания теней издаваемых всеми объектами, выявления возможных недочётов при их моделировании и дальнейшего рендера. Отрендеренная сцена склада с тенями и отражениями изображена на рисунке 3.



Рис. 3. Отрендеренная сцена склада

Выводы

Среда Cinema 4D позволила создать модели объектов, добавить на них текстуры, а также настроить свет, камеру и анимацию для них. Встроенный рендер движок симулировал динамические тени и выдал изображения высокого качества. Ведь Cinema 4D содержит огромное количество инструментов и настроек для создания различных проектов. Начиная от простых геометрических фигур и заканчивая симуляцией частиц, волос и шерсти. В ней существует возможность написания собственных скриптов или плагинов на языке

программирования Python. Также у пользователь может использовать сторонние плагины (RealFlow, TurbulenceFD, Houdini Engine и другие) и рендер движки (Arnold, Octane, V-Ray и другие) от именитых студий.

Среда моделирования Cinema 4D проста в освоении и позволяет дизайнеру самостоятельно моделировать ландшафт, здания, растения, различные объекты и динамику сцены без потребности использования сторонних библиотек, что даёт возможность использовать её для создания иллюстраций, моушн графики, спецэффектов и во многих других сферах. Она заменяет собой множество программ, которые потребовались бы контент мейкеру для реализации поставленных задач и прекрасно вплетена в процесс производства контента за счёт возможности экспорта, импорта и совмещения различных данных.[3]

Литература

1. Ландшафтный дизайн [Electronic resource] / Интернет-ресурс. - Режим доступа : www/ URL: <http://landshaftnyi-dizayn.mydiz.biz/promyshlennyj-landshaftnyj-dizajn>.
2. Cinema 4D Getting Started [Electronic resource] / Интернет-ресурс. - Режим доступа : www/ URL: <http://www.maxon.net/ru/podderzhka/getting-started/>.
3. MAXON Cinema 4D [Electronic resource] / Интернет-ресурс. - Режим доступа : www/ URL: <http://www.maxon.net/ru/industrii/arkhitektura/>

УДК 004.588

ИСПОЛЬЗОВАНИЕ ИГРОВЫХ МЕТОДОВ В ОБУЧАЮЩИХ СИСТЕМАХ ПО РАЗВИТИЮ НАВЫКОВ ПРОГРАММИРОВАНИЯ

Богуцкая В.Б., Губенко Н.Е.

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: vikavolpe@gmail.com, negubenko@mail.ru

Аннотация

Богуцкая В.Б., Губенко Н.Е.. Использование игровых методов в обучающих системах по развитию навыков программирования. Проводится сравнительный анализ ряда концепций применения игровых методов в геймифицированных обучающих системах по программированию, показаны их преимущества и недостатки.

Annotation

Bohutskaya V.B., Gubenko N.E. The use of gaming techniques in training systems for the development of programming skills. A comparative analysis of a number of concepts of the use of gaming methods in gamified training programming systems is carried out, their advantages and disadvantages are shown.

Введение

Концепция использования игровых приёмов в обучении существует уже много лет, но слово «геймификация» довольно новое. Она появилась в 2010 году. Геймификация означает использование игрового мышления, игровых элементов, игровых методологий и игровой среды в неигровом контексте, для повышения заинтересованности пользователей в решении проблем, совершенствования имеющихся и приобретения новых навыков в определённой области знания. Сегодня геймификация широко используется во многих областях, даже для поощрения клиентов и мотивирования сотрудников.

Одной из актуальных сфер использования является образование. Существует сильная связь между геймификацией и обучением. И то, и другое является пассивной напряжённой и утомительной деятельностью и требует мотивации и поощрения от начала и до конца выполняемого процесса. Нет никакой разницы между детьми и взрослыми. Все хотят получать удовольствие от обучения. Поэтому использование игры для обучения взрослых концепциям программирования является актуальным.

Игры должны быть хорошо структурированы и динамичны для того, чтобы побуждать игроков продолжать их. Для стимулирования эмоций, усиливающих мотивацию обычно используются четыре стратегии:

- 1) предоставление возможностей для решения задач, стратегии и решения проблем;
- 2) введение элементов, которые способствуют таинственности, интриге и любопытству;
- 3) ведущие игроки к волнению или рельефным настроениям;
- 4) поощрение конкуренции и командной работы

Цель работы

Существует много типов игр, используемых в обучении. Есть различие между обучением и обучением на основе игры, поэтому целью сравнительного анализа является выявление структурных, адаптивных и других особенностей геймифицированных подходов к обучению.

Геймификация встраивает игровые механики в образование, в то время как игровое

обучение использует игру как часть учебного процесса. Геймификация используется во всех аспектах нашей жизни для увеличения удовольствия и участия, но игровое обучение используется только для обучения.

Существует два основных типа геймификации: структурная и контентная.

Структурная геймификация - это применение игровых элементов для мотивации учащихся через контент без изменения его содержания. Например, чтобы использовать игровые элементы, такие как точки, уровни, значки, списки лидеров и достижения, и применять их в контексте обучения.

Контентная геймификация - это применение игровых элементов, игровой механики и игрового мышления, чтобы изменить контент, чтобы сделать его более похожим на игры. Таким образом, помимо игровых элементов происходит изменение содержания, например, предоставление истории, тайны и персонажей для контента, чтобы вовлечь ученика.

Исследователи выделили два типа академической мотивации - внутреннюю и внешнюю мотивацию. Внутренняя мотивация возникает, когда учащиеся учатся «ради себя», и им это нравится. Вот некоторые примеры: альтруизм, конкуренция, сотрудничество, чувство принадлежности, любовь или агрессия. Внешняя мотивация возникает, когда что-то или кто-то подталкивает ученика к действию, например: классификации, уровни, очки, значки, награды, миссии.

Существует проблема со структурным типом геймификации, которая может уменьшить внутреннюю мотивацию пользователя или заменить его внешней мотивацией, поскольку пользователь фокусируется на получении вознаграждения, а не на действиях. Если же игровые элементы становятся значимыми для пользователя посредством информации, тогда внутренняя мотивация значительно повышается.

Игровая механика или игровые приемы используются для создания блоков геймифицированных приложений (ГП). Награды в виде значков являются одним из распространённых элементов геймификации. Они присваиваются учащимся за выполнение каждого задания. Эта идея заимствована из концепций игр [2]. В некоторых играх есть много значков, назначенных игроку после достижения определенного уровня или после выполнения определенной задачи в игре. Игрок будет счастлив получить этот значок. У студентов будет сильная мотивация выполнять свои задания, чтобы выиграть значки.

Применение методов геймификации

Метод, использующий таблицу лидеров в геймификации позволяет вызвать конкуренцию между студентами в аудитории. В работе «От игровых элементов к геймификации» приведено исследование [2], в ходе которого было выяснено, что студенты усердно работают, потому что каждый хочет увидеть свое имя в таблице лидеров. Мы можем применить этот метод, чтобы побудить учащихся быстро выполнять свои задания. Очки дают студентам положительную мотивацию. Учитель может дать студентам баллы по любой задаче, которую выполняет ученик. Очки геймификации более привлекательны, чем оценки.

Доктор Ян Гловер отмечает, что «средство геймификации стремится к тому, чтобы добавить игру как концепции в процесс обучения» [1]. Это отличается от обучения, основанного на игре и означает создание инструмента, который приносит образовательные преимущества и включает в себя программное обеспечение, такое как симулятор, что даёт значительные преимущества.

Сравнение традиционной модели обучения с геймифицированной приведено в таблице 1.

Таблица 1 – Сравнение традиционной и геймифицированной модели

Традиционная модель	Геймифицированная модель
Учитель учит студентов	Самообучение
Оценки для оценки студентов	Очки для оценки студентов
Текст (книги и доска)	Дизайн (более привлекательный)
Темы для определения содержания курса	Уровни (больше соревнований)
Повышенная сложность	Этапы (сложность)
Текстовое задание	Уровень мастерства
Класс	Ранг

Кандидат наук Ибанез Мария-Бланка использовала геймификацию в обучении языку программирования С для оценки эффективности обучения и вовлеченности [3]. Исследование проводилось с использованием последовательного смешанного метода объяснительных анкет. Система геймификации подготовлена для вопросов студентов и изучения предметов, связанных с языком программирования С, использует основные элементы игровой механики. Результаты показывают, что большинство студентов продолжали работать даже после того, как набрали максимальное количество баллов, чтобы собрать все значки и продолжить обучение.

Кнутас, Иконен и Никула проводили анализ опроса онлайн студентов, которые проходили курс программирования с использованием геймификации. Были выявлены преимущества: улучшение совместной работы студентов, сокращение времени отклика и повышение эффективности обмена информацией [4].

Основной целью исследования Донга, Унча и Чэстина [5], было выявление влияния методов геймификации на взаимодействие и изучение языка программирования С. Была разработана платформа Q-Learning-G. Платформа включает в себя основные элементы игровой механики, где студенты повышают свой уровень знаний, представляя и оценивая вопросы, связанные с программированием.

Редфилд [6] имеет опыт использования техники геймификации в обучении студентов основам компьютерного программирования. В этой модели студенты играют, оценивают и представляют, демонстрируют и создают свои игры. Использование игр в образовании способствует ускорению процесса обучения, но для этого требуется отличный инструктор по разработке игр. Это может быть эффективно применено к процессу обучения

Бутгерейт [7] использует геймификацию со взрослыми в интернет-модуле, чтобы поощрить студентов, которые очень заняты и имеют мало свободного времени. В этом примере используется структурная геймификация, потому что автор использовал механики геймификации: краткосрочные цели, очки, списки лидеров, значки. Использование этих механизмов показывает положительные результаты.

Примеры эмпирических исследований ряда существующих геймифицированных систем обучения программированию, посвящённых изучению геймификации для обучения концепциям программирования были проведены и представлены в таблице 2 [2].

Таблица 2 – Примеры исследований

Методы, используемые в исследовании	Тип применения	Год	Страна	Предмет	Преимущества	Игровая механика
Анкета и статический анализ [3]	Активированный подход	2014	Испания	Язык программирования С	Эффективность и участие	Точки Значки
Опрос студентов [4]	Онлайн-студент	2014	Малайзия	Введение в программирование	Сотрудничество, сокращение времени отклика	Точки Значки
Анкета и статический анализ [5]	Активированный подход	2012	Пенджаб	Концепции и программирования	Мотивационное взаимодействие	Точки Значки
Реализация [6]	Традиционный подход	2013	США	Разработка игр	Веселье	Раскадровка

На основе анализа литературных источников была разработана геймифицированная обучающая система по математике для начальной школы на основе ментальных карт с использованием объектно-ориентированного подхода. Диаграмма вариантов использования разработанной обучающей системы представлена на рисунке 1.



Рисунок 1 – Диаграмма вариантов использования

В целях увеличения заинтересованности обучаемых в систему был добавлен персонаж «Сова» и очки в виде звёздочек, характеризующие количество баллов за выполнение задания, представлены на рисунке 2.



Рисунок 2 – Персонаж обучающей системы и очки за выполнение задания

Выводы

Как показал анализ литературных источников геймификация дает множество эмоций, таких как сильные эмоции, от любопытства к разочарованию до радости, позитивные эмоциональные переживания, такие как счастье и гордость, повышает уровень взаимодействия и мотивации в классах, которые непосредственно необходимы студентам. Социальное взаимодействие между учащимися через разговоры, слушание и активное изучение концепций происходит, когда учащиеся сталкиваются с проблемой, они участвуют в обсуждении с целью поиска решений.

Была разработана модель геймифицированной обучающей системы по математике. Модель предполагает дальнейшее развитие в программный комплекс, предназначенный для заочных форм обучения детей начального школьного возраста, для получения начальных знаний математики.

Литература

1. Glover, I. Play as You Learn: Gamification as a Technique for Motivating Learners [Текст]/Glover I., Herrington J., Couros A., Irvine V. // Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2013. - Chesapeake: AACE, 2013. - С. 1999-2008.
2. Deterding, S. From Game Design Elements to Gamefulness: Defining Gamification [Текст] / Deterding S., Dixon D., Khaled R., Nacke L. // Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments 2011. - New York: ACM, 2011. - С. 9-15.
3. Ibanez, M. B. Gamification for Engaging Computer Science Students in Learning Activities: A Case Study / Ibáñez M. B., Di-Serio A., Delgado-Kloos C // IEEE Transactions on Learning Technologies. - 2014. - No 7.-С. 291-301.
4. Knutas, A. Increasing Collaborative Communications in a Programming Course with Gamification: A Case Study. / Knutas A., Ikonen J., Nikula U., Porras J. // International Conference on Computer Systems and Technologies. - 2014. - No 14.-С. 1-8.
5. Dong, Z. Engaging Computer Science Students through Gamification in an Online Social Network Based Collaborative Learning Environment. / Dong Z., Untch R.H. and Chasteen M. // IJNET. - 2014. - No 3.-С. 72-77.
6. Redfield, C.L. Gamification and Creating Game Developers [Текст] / Butgereit, L. // Proceedings of the Information Systems Educators Conference 2013. - San Antonio:St. Mary's University Computer Science, 2013. - С. 1-13.

УДК 004.588

СОЗДАНИЕ ИНФОРМАЦИОННОГО САЙТА С ВОЗМОЖНОСТЬЮ СОСТАВЛЕНИЯ ИНДИВИДУАЛЬНОЙ ПРОГРАММЫ ТРЕНИРОВОК

Катков М.В., Павлий В.А

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: aia2003@ukr.net

Аннотация:

Катков М.В., Павлий В.А Создание информационного сайта с возможностью составления индивидуальной программы тренировок. Рассмотрена задача создания информационного сайта о занятии спортом и здоровом образе жизни. В проекте использована технология WordPrees 5.1.1. Основываясь на анализе сайтов-аналогов, предложен ряд возможностей сайта – просмотр информации о готовых спортивных тренировках и спортивных программах, а так же информации о здоровом образе жизни, видео материалов; зарегистрированным пользователям предоставлена возможность составить личную программу тренировок на основе полученных от него данных, а также просматривать ее в личном кабинете, комментировать инфо-материалы сайта. Созданный сайт дает возможность заинтересованным посетителям быть в курсе информации о спорте и облегчает поиск и составление личной программы тренировок.

Annotation:

Katkov M. V., Pavly V. A Creation of an information site with the possibility of drawing up an individual training program. The problem of creating an information site about sports and healthy lifestyle is considered. The project uses WordPrees 5.1.1 technology. Based on the analysis of sites-analogues, a number of features of the site – view information about ready-made sports training and sports programs, as well as information about healthy lifestyles, video materials; registered users are given the opportunity to create a personal training program based on the data received from him, as well as view it in your account, comment on the info-materials of the site. The created site allows interested visitors to be aware of information about the sport and facilitates the search and preparation of a personal training program.

Общая постановка проблемы

В последние годы веб технологии предоставляют возможность выйти за рамки типичного предоставления пользователю контента, позволяя установить с ним диалог и давая возможность предоставить ему необходимую информацию основываясь на его персональных данных.

Введение

Спорт — организованная по определённым правилам деятельность людей, состоящая в сопоставлении их физических и (или) интеллектуальных способностей, а также подготовка к этой деятельности и межличностные отношения, возникающие в её процессе.

Спорт представляет собой специфический род физической или интеллектуальной активности, совершаемой с целью соревнования, а также целенаправленной подготовки к ним путём разминки, тренировки. В сочетании с отдыхом, стремлением к постепенному улучшению физического здоровья, повышению уровня интеллекта, получению морального удовлетворения, к совершенству, улучшению личных, групповых и абсолютных рекордов, славе, улучшению собственных физических возможностей и навыков спорт предназначен для совершенствования физико-психических характеристик человека.

Массовый спорт даёт возможность миллионам людей совершенствовать свои физические качества и двигательные возможности, укреплять здоровье и продлевать творческое долголетие.

Цель спорта высших достижений — это достижение максимально возможных спортивных результатов или побед на крупнейших спортивных соревнованиях.

Основным назначением сайта является оперативное и полное предоставление информации для всех групп потенциальных пользователей. Для обеспечения эффективного выполнения этих функций в современных условиях необходимо решить следующие задачи:

- провести анализ современных систем управления содержимым сайта и выбрать оптимальную для использования;
- провести анализ сайтов-аналогов, и на основе полученных результатов разработать структуру разделов сайта;
- проектировать макет страницы сайта в соответствии с его структурой, функциями и психологическими особенностями восприятия человеком информации;
- спроектировать структуру групп пользователей, а так же права их доступа к разделам и управлению содержимым сайта.

Анализ современных средств реализации

Большинство современных интернет-сайтов реализовано с помощью систем управления содержимым (англ. Contentmanagementsystem, CMS) – компьютерных программ, которые предназначены для обеспечения и организации совместного процесса создания, редактирования и управления содержимым сайта.

Использование системы управления содержимым даёт целый ряд преимуществ:

- простота редактирования содержимого сайта с использованием современных визуальных редакторов; это даёт возможность оперативно и качественно редактировать содержимое пользователями, не владеющими знаниями языков программирования или гипертекстовой разметки;
- расширяемость за счёт установки дополнительных модулей;
- управление содержимым пользователями с различным уровнем доступа;
- автоматическое обновление и следование веб-стандартам.

Среди современных систем управления содержимым интернет-сайтов можно выделить три самых распространённых: WordPress, Joomla и Drupal. Система WordPress чаще всего используется для управления новостными сайтами и блогами. Систему Joomla целесообразно использовать для управления сайтом с богатой и гибкой структурой, используя расширенные функции системы. Для управления сайтами со сложной организацией данных, например таких, как интернет-магазин, чаще используют систему Drupal.

Сайт группы имеет преимущественно информационный характер, и соответственно должен иметь разнообразную и гибкую структуру содержимого. Выполненный анализ показал, что для его модернизации целесообразно использовать систему WordPress.

Анализ структуры сайтов-аналогов

Для анализа были выбраны сайты из первых позиций по запросу “Составление программы тренировок”, “Спортивные тренировки”, “Спорт” в поисковых системах Yandex, Google, представленные в таблице 1. Для данных сайтов был проведён детальный анализ их структуры.

В качестве наиболее характерных были выделены следующие основные разделы для структурирования содержимого сайта: Главная страница, статьи, тренировки, составить тренировку, вход, регистрация, контакты.

Таблица 1. Структуры разделов сайтов

Группа	Структура
sportwiki.to	Главная, о спортвики, все статьи, новые статьи, случайная статья, свежие правки, категории, цели, справка инструменты, эксперты, обратная связь, вход, регистрация.
tvoytrener.com	Дневник, тренировки, упражнения, диеты, калькуляторы, статьи, спросить, анкеты, вход, регистрация.
hvat.ru	Фитнес, бодибилдинг, кроссфит, workout, единоборства, бег, разное, о нас, вход, регистрация.
wefit.ru	Главная, тренировки, упражнения, питание, мотивация, медицина, спорт, новости, вход, регистрация.

Таблица 2. Общий анализ аналогов сайтов

Сайт	Дизайн	Реклама	Контент	Адаптивность	Доп. возможности
sportwiki.to	+	-	+	-	+
tvoytrener.com	+	-	+	-	+
hvat.ru	+	-	+	+	-
wefit.ru	+	+	+	+	-

Особенности организации сайта

К базовым элементам отдельных страниц сайта относят заголовок страницы, навигационное меню, ленту содержимого и подпись организации. Как правило, подпись располагают внизу после содержимого страницы. Для представления ссылок на ресурсы, зачастую используют баннеры. Их целесообразно группировать в отдельные блоки интернет страницы. Навигационное меню является важной частью страницы сайта и его расположение должно не только привлекать внимание нового посетителя, но и быть удобным для постоянного посетителя. Следовательно, целесообразно разместить два типа связанных навигационных меню, в верхней части страницы и на уровне ленты содержимого для более удобного использования.

Учитывая особенности физиологии и психологии восприятия человеком информации, просмотр содержимого страницы им осуществляется в соответствии с правилами чтения (в большинстве стран мира – слева-направо, сверху-вниз). Следовательно, располагать основные элементы страницы необходимо по аналогии в порядке их важности. Разработанный макет структуры страницы интернет сайта представлен на рисунке 1.

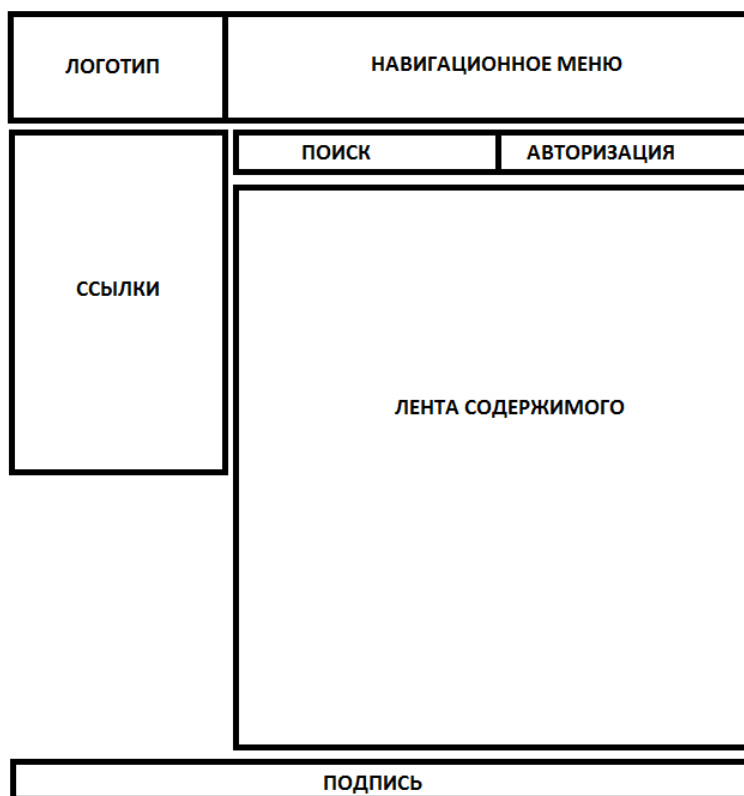


Рисунок 1 – Макет страницы интернет сайта

Все пользователи сайта делятся на зарегистрированных и незарегистрированных. Не зарегистрированные пользователи имеют доступ ко всем инфо-материалам сайта, но не имеют возможность комментирования статей и не имеют свой личный кабинет. Зарегистрированные пользователи сайта имеют доступ к личному кабинету в котором будут храниться данные о составленной сайтом для них тренировки а также имеют возможность оставлять комментарии к статьям. Администратор имеет все права. Для организации прав доступа к разделам сайта и возможностей редактирования его содержимого, необходимо создать следующие категории прав доступа: Посетитель, Участник, Администратор.

Реализация сайта

Для реализации сайта был проведен анализ (таб. 3) и выбрана система управления содержимым (CMS) WordPress версии 5.1.1. Данная версия считается в настоящее время основной и наиболее стабильной, поддерживает официальные пакеты русификации и современные версии языков программирования и разметки PHP, JavaScript, HTML и CSS. Редактирование и настройка компонентов сайта будет осуществлена средствами панели управления CMS. В данном приложении осуществляется настройка всех модулей сайта, загрузка дополнительных расширений, выбор и корректирование шаблонов оформления, управление пользователями и все возможные изменения.

Таблица 3 – Сравнение систем управления содержимым (CMS)

	Joomla	WordPress	1С-Битрикс	Drupal
Цена	-	-	+	-
Удобство для контент-менеджера	-	+	+	-
Удобство для	-	-	+	-

разработчика				
Встроенные модули	+	+	+	+
Ассортимент шаблонов дизайна	+	+	-	-
Пригодность для SEO-оптимизации, изначальная оптимизированность	-	+	+	-
Скорость работы	-	+	+	-

С помощью панели управления CMS будут реализованы компоненты меню сайта, в соответствии с проектом структуры, добавлена группа баннерных ссылок на дополнительные ресурсы, настроен модуль авторизации пользователей, добавлен модуль поиска по сайту, реализован проект группирования пользователей и их доступа к содержимому сайта. В соответствии с проектом макета страницы будут расположены все базовые компоненты, будет настроен стандартный шаблон оформления, реализована возможность представления содержимого сайта на двух языках, а именно английском, русском. В процессе редактирования и наполнения сайта, системой управления заполняется база данных MySQL, которая хранит все используемые модули и компоненты сайта, их настройки и добавленные материалы.

Выводы

На основе анализа современных способов управления интернет сайтом и структуры сайтов-аналогов разработан проект информационного сайта с возможностью составления личной программы тренировок, который реализован на базе системы управления содержимым WordPress версии 5.1.1.

На сайте планируется реализация системы поиска по содержимому, конструктор тренировок, модуль авторизации пользователей, статьи.

Литература

1. Атлетический тюнинг. Сослан Варзиев
2. Новая энциклопедия бодибилдинга. Арнольд Шварценеггер
3. 10 самых популярных CMS. <https://webformyself.com/10-samyx-ispolzuemyx-cms-sistem-upravleniya-kontentom/>
4. Создание сайтов на основе WordPress. Алексей Сергеев. Донецк: ДонНТУ, 2019.

УДК 004.588

ЭЛЕКТРОННЫЕ БИБЛИОТЕКИ И ПОТРЕБНОСТИ ПОЛЬЗОВАТЕЛЕЙ. ИССЛЕДОВАНИЕ РЕКОМЕНДАЦИЙ ПО ДИЗАЙНУ ЭЛЕКТРОННЫХ БИБЛИОТЕК И РАЗРАБОТКА ТРЕБОВАНИЙ НА ИХ ОСНОВЕ

Донцова Е.А

Донецкий национальный технический университет
кафедра компьютерного моделирования и дизайна

E-mail: doncova-98@mail.ua

Аннотация:

Донцова Е.А. Электронные библиотеки и потребности пользователей. Исследование рекомендаций по дизайну электронных библиотек и разработка требований на их основе. В статье рассмотрена актуальность разработки электронной библиотеки. Приведен краткий обзор первых электронных библиотек. Выявлены преимущества и недостатки данной системы. Разработаны основные требования для получения эффективного и качественного продукта.

Annotation:

Dontsova E.A. The article discusses the relevance of developing an electronic library. A brief review of the first electronic libraries is given. Identified the advantages and disadvantages of this system. Developed the basic requirements for an effective and quality product.

Введение

Регулярное увеличение информации требует новых методов ее сбора, обработки, хранения и передачи. Применение и анализ информации являются ключевыми факторами, характеризующими степень развития общества. После изобретения книгопечатания, основным центром, обеспечивающим население информационными ресурсами, стали библиотеки. Достижения научно-технического прогресса привели к решению проблемы создания хранилищ для больших объемов данных и способов предоставления доступа к ним любым категориям граждан. Так впервые появилось понятие «цифровая» или «электронная» библиотека. Поскольку данные хранилища могут предоставлять обзор разнородных документов через подключение к интернету в любой момент и имеют эффективные механизмы поиска материалов, электронная библиотечная система приобрела значительную популярность.

Первые электронные библиотеки

Первая идея перевода печатного слова в цифровой вид принадлежит американцу Майклу Харту в 1971 году. Основанный им проект получил название «Гутенберг» и является общественной инициативой по оцифровке, архивированию и распространению произведений мировой литературы. На данный момент в каталоге проекта насчитывается более 50 000 бесплатных книг. [1] Сайт имеет удобный и простой интерфейс и хорошо организованную структуру (рис.1) Предоставляет возможность пользователям отправлять электронные книги в Dropbox и Google Drive, получать новые записи через RSS-канал или по электронной почте, находить старинные изображения из книг.



Рисунок.1. Библиотека Гутенберг

Первое русскоязычное книгохранилище в Интернете появилось в 1994 году и было основано Максимом Мошковым. На данном ресурсе представлены самые разнообразные произведения художественной, документальной и учебной литературы. Архивы пополняются главным образом усилиями читателей. [2] Из достоинств можно выделить простейший интерфейс, удобная навигация, бесплатный доступ к текстам. К недостатку можно отнести примитивный дизайн, который не менялся с момента создания. Главная страница располагается на белом фоне, на котором в виде ссылок перечислены разделы книг (рис.2). При длительном просмотре очень устают глаза, что затрудняет чтение.

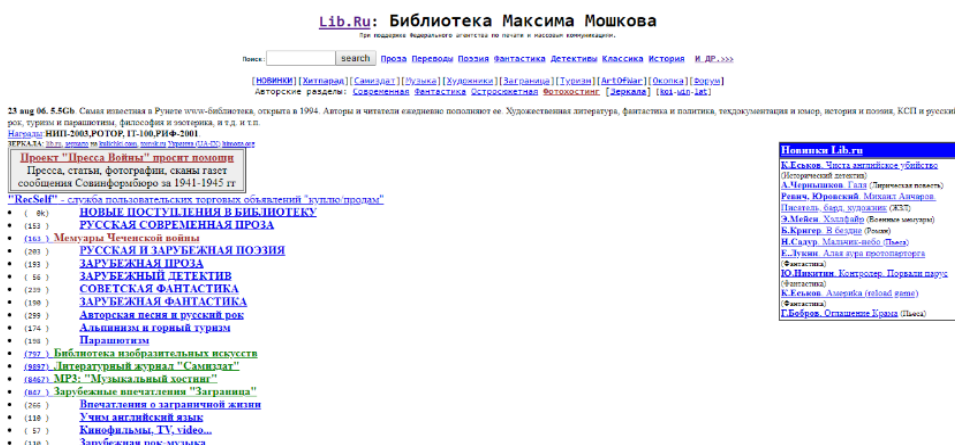


Рисунок.2. Библиотека Максима Машкова

Достоинства и недостатки электронных библиотек

При принятии решения о том, стоит ли создавать электронную библиотеку или нет, необходимо учесть некоторые её преимущества и недостатки. К преимуществам можно отнести следующие:

- Возможность пользоваться литературой в любое время и в любом месте, пока доступно подключение к сети Интернет. Это является отличным способом обучения.
- Онлайн-библиотеки помогают научному сообществу, поскольку они служат местом для хранения важных исследований, диссертаций и т.д.
- Доступ к редким изданиям.
- Многие книги можно скачать, продублировать и перенести на электронный носитель.
- Для более удобного поиска, каждая онлайн-библиотека оснащена системой поиска.

- Доступно для людей с ограниченными возможностями.
- Оцифровка может улучшить разборчивость и устранить видимые дефекты, такие как пятна и обесцвечивание.
- Нет нужды получать читательский билет, возвращать книги в срок и бояться их повредить.

Существуют некоторые ограничения и недостатки:

- Дорогостоящие устройства для просмотра электронных ресурсов.
- Чтение с экранов компьютера, телефона или планшета вредит зрению и здоровью человека.
- Ограничение законом об авторском праве.

Большинство хранилищ оцифрованной литературы организованы в виде web-сайтов. Этот способ является наиболее удобным, так как для получения доступа к необходимой публикации достаточно подключиться к сети Интернет и воспользоваться браузером.

Требования к разрабатываемому продукту

Наиболее целесообразным решением для разработки электронной библиотеки будет использование сценарного языка программирования PHP для работы с серверной составляющей приложения, языка разметки гипертекстовых страниц (HTML) для создания интерфейса сайта, таблицами каскадных стилей для создания внешнего оформления (CSS) и системой управления базами данных (MySQL).

Для упрощения процесса разработки web-сайта имеется возможность воспользоваться системой по управлению контентом (CMS).

Основные функции CMS:

- предоставление инструментов для создания содержимого, организация совместной работы над содержимым;
- управление содержимым: хранение, контроль версий, соблюдение режима доступа, управление потоком документов;
- публикация содержимого;
- представление информации в виде, удобном для навигации, поиска. [3]

Наиболее популярной системой на сегодняшний день является WordPress. По результатам исследования технологической лаборатории W3Techs в 2018 году около 30% всех существующих в мире веб-страниц построены на WordPress. Среди всех сайтов, использующих CMS, доля WordPress достигла 60,2%. [4]

Готовый продукт должен быть удобен в использовании, полезен и привлекателен для посетителей. Для выполнения этих условий нужно придерживаться некоторых рекомендаций:

- Поместить ссылку “Помощь” на каждой странице.
- Использовать только одну панель навигации.
- Выделять другим цветом или подчеркивать ссылки, которые пользователь уже посетил.
- Организация материалов по темам. Многие пользователи, заходя на библиотечные сайты, знают категорию контента, которую они хотят найти. Например, студенты часто ищут статьи из журналов, потому что они могут быть полезнее для изучения, чем художественные произведения или справочные материалы. Компоновка всех ресурсов по типу материала поможет существенно упростить поиск нужных сведений.
- Простой поиск по автору, названию и ключевым понятиям(рис.3).

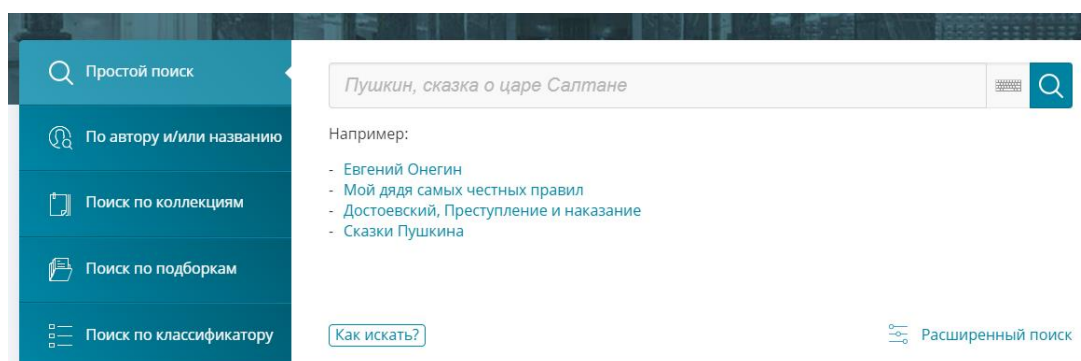


Рисунок.3. Удобный поиск по автору, названию,коллекциям,подборкам и классификатору

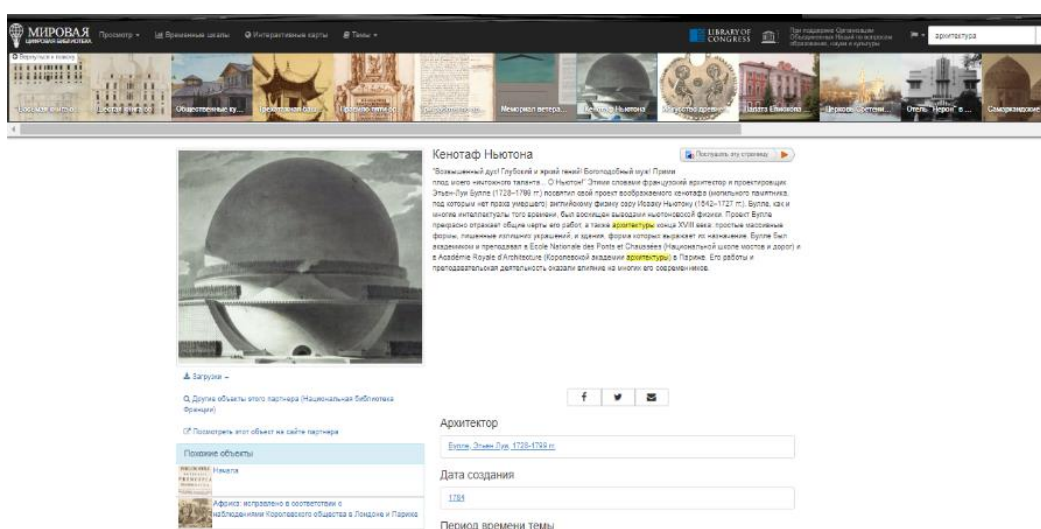


Рисунок.4. Основное содержание занимает большую часть страницы, панель навигации и логотип расположены вверху.

– Избегать или свести к минимуму анимацию текста, так как многие пользователи считают это отвлекающим и раздражающим. Не допускается применение более трех различных шрифтов. Для выбранной стилистики хорошо подойдет Open Sans. Нейтральный и легкий для восприятия шрифт, который удобен для чтения и прекрасно сочетается со многими другими шрифтами.

– Важно правильно подобрать дизайн всех страниц в одном стиле. Основное содержание должно занимать больше места. Логотип и панель навигации занимают верхнюю часть сайта(рис.4).

– Не злоупотреблять иллюстрированными материалами, аудио- и видеофрагментами.

– Отдельное внимание нужно уделить выбору цветовой гаммы. Правильно подобранные цвета и дружелюбный интерфейс – самые мощные инструменты привлечения и удержания внимания. Дизайнеры советуют не увлекаться обилием цветов на странице. Обычно лучше выбрать один главный цвет, который компенсируется нейтральным, таким как белый, серый или черный. Стоит избегать неоновых цветов. Они чрезмерно яркие и отвлекают от основного посыла. При выборе цветовой палитры для электронной библиотеки лучше остановиться на более мягких и спокойных тонах. Хорошо

подойдут белый, светлые оттенки синего, зеленого и коричневого. Не стоит применять фоновый рисунок, это затрудняет восприятие текста. Чтобы сосредоточить внимание на отдельной части веб-сайта, например, навигационной системе, следует придать ей более насыщенный цвет.

– Проверить сайт на удобство использования и убедиться, что он отвечает всем потребностям читателей.

Выводы

В результате проведенных работ была рассмотрена история появления и развития электронных библиотек. Так же был обоснован выбор средств разработки электронного хранилища оцифрованной литературы, а именно CMS Wordpress, как наиболее распространенной. Также были разработаны требования на основе рекомендаций специалистов по дизайну web-сайтов для повышения комфортности использования данного ресурса.

Литература

1. Проект «Гутенберг» [Электронный ресурс]: [сайт проекта]. - [Б. м., 20--]. - Режим доступа: <http://www.gutenberg.org> (Дата обращения 04.05.2019)
2. Библиотека Максима Мошкова [Электронный ресурс]. - [Б. м., 1994-]. -Режим доступа: <http://www.lib.ru> (Дата обращения 05.05.2019)
3. Савельева Н. Системы управления контентом (рус.) // Открытые системы. — 2004. — № 4. (CMS)
4. World Wide Web Technology Surveys [Electronic resource] / Интернет-ресурс. - Режим доступа: [www/ URL: https://w3techs.com](http://www.w3techs.com) (Дата обращения 05.05.2019)

СЕКЦИЯ 8

КОМПЬЮТЕРНАЯ ИНЖЕНЕРИЯ

ИУС
МКМ 2019

УДК 004.3

АНАЛИЗ ТЕСТИРУЕМОСТИ КОНСТАНТНЫХ НЕИСПРАВНОСТЕЙ НА БАЗЕ T-МОДЕЛИ ОБЪЕКТА ДИАГНОСТИКИ

Лактионов А. А., Зинченко Ю. Е., Зинченко Т. А.

Донецкий национальный технический университет, кафедра компьютерной инженерии

E-mail: sangangstar@gmail.com

Аннотация:

Лактионов А. А., Зинченко Ю. Е., Зинченко Т. А. Анализ тестируемости константных неисправностей на базе t-модели объекта диагностики. Разработана структура имитационного моделирования, ориентированная на решение задачи анализа тестируемости константных неисправностей цифровых устройств на основе T-модели объекта диагностики. Приводится функциональная схема анализатора тестируемости и описывается его функционирование. Описывается также реализация подсистемы моделирования неисправностей в режиме анализатора тестируемости на языке описания аппаратуры VHDL. Полученные результаты могут быть использованы в процессе контролепригодного проектирования БИС и в учебном процессе.

Annotation:

Laktionov A. A., Zinchenko Yu. E., Zinchenko T. A. Analysis of testability of constant faults based on the t-model of the object of diagnosis. The structure of simulation modeling has been developed, focused on solving the problem of analyzing the testability of constant faults of digital devices based on the T-model of the diagnostic object. A functional diagram of the testability analyzer is given and its functioning is described. The implementation of the fault simulation subsystem in the testability analyzer mode in the VHDL hardware description language is also described. The results obtained can be used in the process of controllable LSI design and in the educational process.

Введение

Традиционный подход решения большинства задач тестового диагностирования (ТД) ДУ состоит в применении для каждой из них индивидуального подхода, начиная от идеи и заканчивая технической реализацией [1]. Это приводит к высокой трудоемкости и, в конце концов, к высокой стоимости диагностического обеспечения. В то же время общая природа этих задач создает почву для унифицированного подхода. С другой стороны в последнее время на рынке вычислительной техники получили стремительное развитие САПРы с функцией имитационного моделирования. В этой связи особенно актуальным становится разработка унифицированного подхода для решения диагностических задач на основе имитационного моделирования[2-4].

1 T-модель дискретного устройства

На кафедре компьютерной инженерии ДонНТУ разрабатывается методология решения диагностических задач цифровых устройств (ЦУ) на основе так называемой T-модели объекта диагностики (ОД) [5].

В общем случае ОД дискретного типа со множеством входов $X=\{x_i\}$, $i=1,\dots,n$, выходов $Y=\{y_j\}$, $j=1,\dots,m$, множеством моделируемых неисправностей $H=\{h_k\}$, $k=1,\dots,p$ и его T-модель можно представить обобщенными структурами:

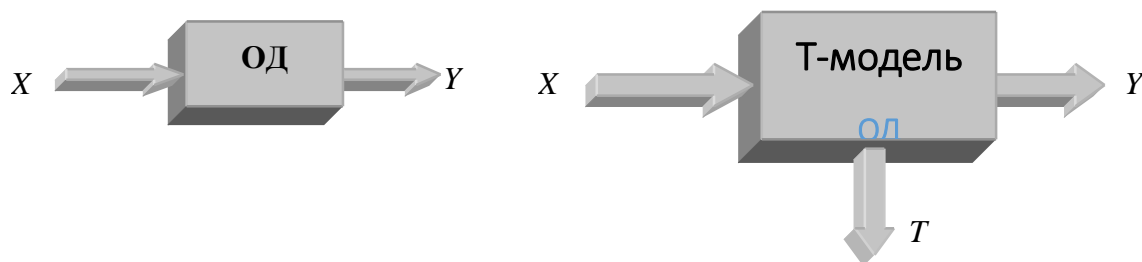


Рисунок 1 – Дискретный ОД и его Т-модель

Особенностью Т-модели является наличие специальных выходов $T=\{T_k\}$, называемых выходами тестируемости неисправностей ОД. Выход T_k выражает условие обнаружения неисправности h_k входной тестовой последовательностью на выходах Y и может быть описан функцией тестируемости неисправности $T(h_k)$, которая в общем случае определяется над входами и внутренним состоянием ОД.

Принцип решения диагностических задач на базе Т-модели отображается структурой, представленной на рис. 2. Такая структура кроме собственно Т-модели включает в свой состав Т-оболочку, функции и схема которой определяется типом решаемой задачи. Используемый в структуре генератор случайных либо псевдослучайных тестов (ГСТ) используется для возбуждения функций тестируемости по случайному (псевдослучайному) закону. В дальнейшем условимся, говоря о случайном тесте либо генераторе иметь ввиду также и псевдослучайные их разновидности. Под воздействием теста на выходах тестируемости $\{T(h_k)\}$ появляется тестовая реакция, так что если в некоторый момент времени t на входах ОД генератор тестов выставляет тест неисправности h_k , то соответствующий выход тестируемости взводится в лог. “1”; в противном случае на этом выходе лог. “0”. Тестовая реакция поступает на Т-оболочку, специфическую для каждой задачи, которая на своем выходе \mathcal{R} выставляет результат решения задачи.

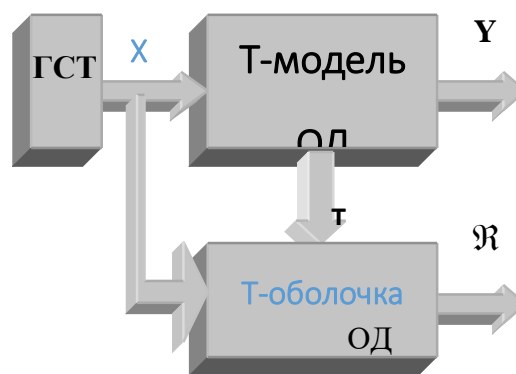


Рисунок 2 - Принцип решения диагностических задач на базе Т-модели

Спектр задач, которые могут быть решены представленным подходом, может быть достаточно широк. Авторами данной работы решается задача анализа тестируемости КН на базе Т-модели ОД.

Структура подсистемы моделирования неисправностей в режиме анализатора тестируемости

Подсистема моделирования неисправностей (ПМН) в режиме анализатора тестируемости предназначена для вычисления тестируемости одиночных константных неисправностей логических схем. Ее структура показана на рис. 3. В ее состав входят генератор псевдослучайного теста (ГПСТ), Т-модель объекта диагностики и анализатор тестируемости (АТ).

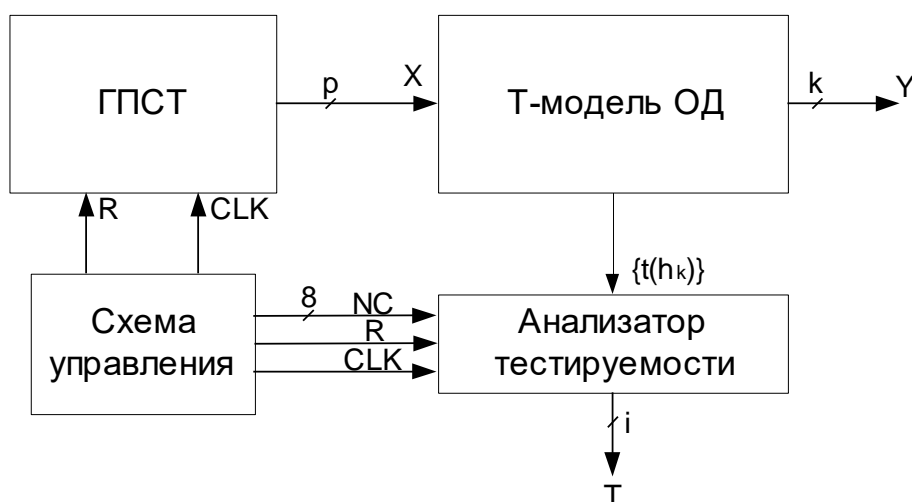


Рисунок 3 - Структура ПМН в режиме анализатора тестируемости

Как видно из рисунка, генератор тестов имеет n -выходов X , которые подаются на функциональные входы Т-модели ОД. Т-модель в свою очередь имеет два типа выходов: Y и t . Выходы Y имеют разрядность m и являются обычными функциональными выходами ОД. Кроме них Т-модель содержит специальные выходы t , которые называются выходами тестируемости и являются неотъемлемой частью любой Т-модели, так как они реализуются логическими элементами, на которых строится модель. Их число соответствует количеству моделируемых одиночных КН и меняется от модели к модели. Задачей анализатора тестируемости является подсчет единиц на выходах тестируемости Т-модели и выдача результата для оценки проектировщиком ЦУ. Схемная реализация подсистемы в режиме анализатора тестируемости приведена на рис. 4.

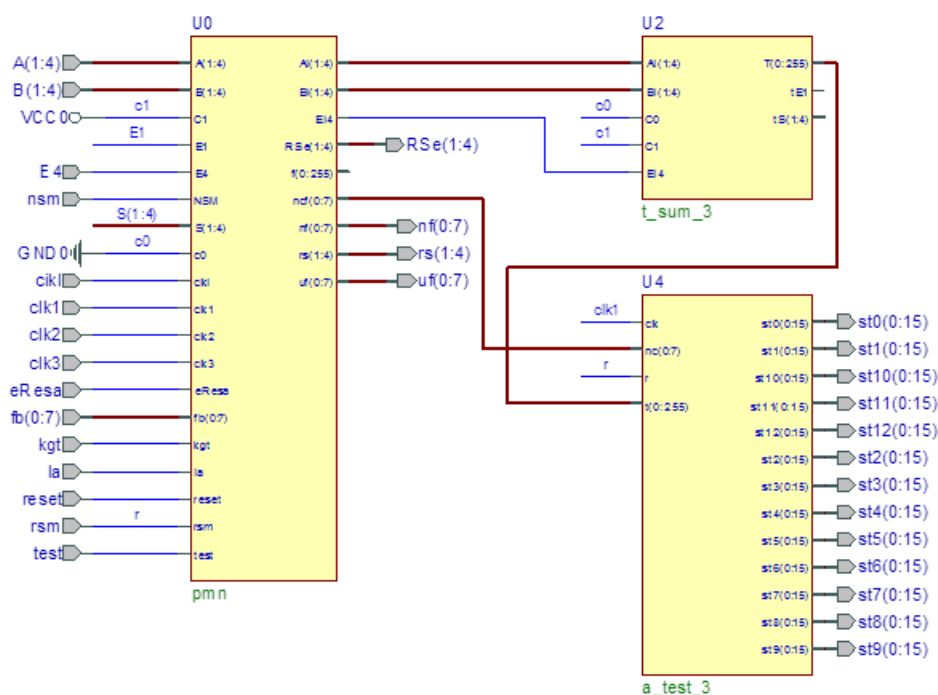


Рисунок 4 - Схемная реализация подсистемы в режиме анализатора тестируемости

На рис. 4 можно увидеть, что выходы элемента PMN AI, BI и EI4 подаются на входы T-модели сумматора, а выход NCF на вход NC анализатора тестируемости. Здесь стоит отметить, что моделируются 4-разрядные сумматоры с последовательным переносом и именно поэтому выходы AI и BI имеют разрядность 4 бита, как и входы T-модели A и B. Выход EI4 представляет собой перенос, подаваемый на младший разряд сумматора. Сигнал NCF, и соответственно NC, имеет разрядность 8 бит и содержит номер цикла моделирования. Число неисправностей не должно превышать 256 (0-255). Кроме этих входов AT имеет входы синхронизации CLK и сброса R.

Выходами анализатора являются выходы счетчиков, которые на схеме обозначены как ST(j). Счетчики являются 16-разрядными и тем самым позволяют хранить значение до 65535.

Анализатор тестируемости

Одной из задач данного проекта является разработка анализатора тестируемости. Понятие тестируемости было довольно подробно рассмотрено в предыдущей главе. Анализатор должен подсчитывать количество единиц на выходах тестируемости, а инженер-проектировщик должен использовать полученные данные для оценки тестируемости тех или иных неисправностей.

В простейшем случае анализатор тестируемости представляет собой линейку из счетчиков, число которых совпадает с числом моделируемых неисправностей. Такое решение является идеальным с точки зрения определения тестируемости каждой неисправности, но на практике это связано с рядом проблем. Во-первых, число неисправностей достаточно велико даже для небольших схем. Например, 4-разрядные сумматоры с последовательным переносом, которые будут рассмотрены позже, насчитывают от 98 до 190 неисправностей.

Довольно трудно представить элемент с 98 выходами. Во-вторых, большое количество счетчиков сильно отразится на времени моделирования ОД. В-третьих, построение такого AT может потребовать много времени. Это потребовало поиск компромиссного решения. Для уменьшения числа счетчиков было решено использовать каждый счетчик для подсчета сразу нескольких выходов тестируемости путем мультиплексирования. Таким образом, предлагается две схемы анализатора тестируемости: параллельная (рис. 5 (а)) и параллельно-последовательная (рис. 5 (б)).

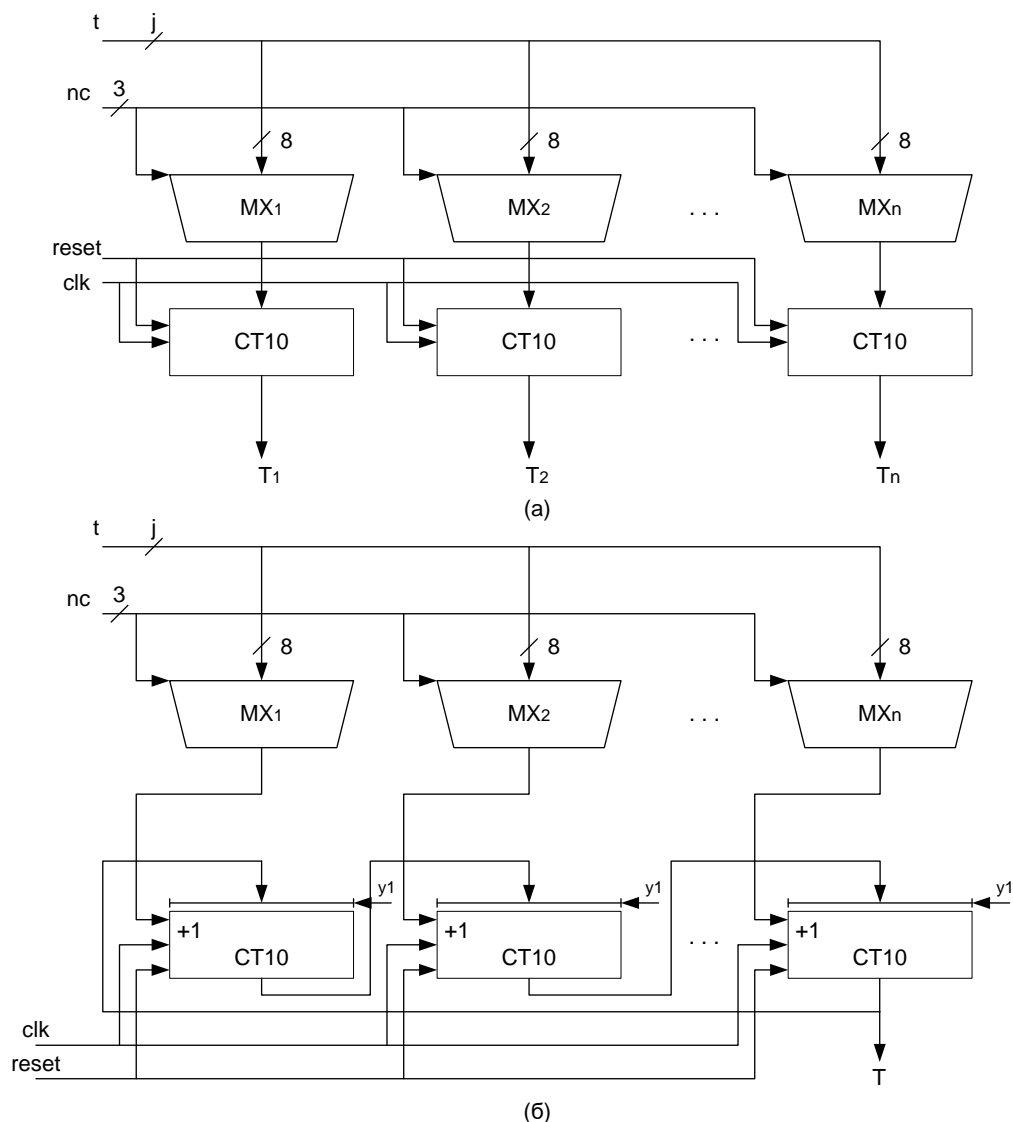


Рисунок 5 - Функциональная схема анализатора тестируемости

Для исследования была разработана экспериментальная ПМН, с помощью которой был выполнен анализ тестируемости КН для суммирующих схем различных типов.

Выводы

На основе предложенной структуры имитационного моделирования возможно решение конкретных диагностических задач. В их числе поиск труднотестируемых неисправностей и расчет параметров (длины и вероятности обнаружения неисправностей) случайного теста по концепции «труднотестируемой неисправности»; определение среднестатистической КН ОД и расчет параметров случайного теста по концепции «среднестатистической неисправности»; определение показателей вероятностной тестируемости отдельных КН и интегрированного показателя вероятностной тестируемости ДУ в целом; определение параметров случайного (псевдослучайного) теста, близких к оптимальным на основе среднестатистической КН.

Разработано экспериментальное ПО на языке С и языке программирования аппаратуры VHDL. Выполнены экспериментальные исследования полученных результатов в

САПР ACTIVE-HDL на примере решения задачи анализа тестируемости КН для суммирующих схем различных типов. ПО внедрено в учебный процесс кафедры компьютерной инженерии ДонНТУ в лабораторный практикум и курсовой проект по "Верификация и контролепригодное проектирования компьютерных систем».

В целом предложена новая структура и подход имитационного моделирования, ориентированная на решение задачи анализа тестируемости константных неисправностей цифровых устройств на основе Т-модели объекта диагностики, что может найти применение при контролепригодном проектировании БИС, в том числе ПЛИС типа FPGA и CPLD.

Литература

1. Abramovici, Miron Digital system Testing & Design. – IEEE, 1990. – 570 p.
2. Зинченко Ю.Е., Жилин К.Е., Козинец А.М., Алексеев П.С. Глазков Д.Е. Моделирование диагностических задач в научных исследованиях и учебном процессе/ Международная научно-методическая конф., 1999г., Днепродзержинск: ДГТУ. С 165-166.
3. Зинченко Ю.Е., Жилин К.Е., Алексеев П.С., Глазков Д.Е.. Диагностирование дискретных устройств на основе имитационного моделирования // Сб. научн. трудов ХТУРЭ: ХТУРЭ.- 1999г., С 251-253.
4. Зинченко Ю.Е. Козинец А.М, Жилин К.Е. Проблемы зондового поиска неисправностей и пути их разрешения / Сб. тр. ДонГТУ. Сер.: Информатика, кибернетика и вычислительная техника, вып. 6. – Донецк: ДонГТУ, 1999.- С. 212--217.
5. Зинченко Ю.Е. Т-модель дискретного устройства и решение диагностических задач на ее основе // Вестник ТРТУ – ДонГТУ. Материалы II Межд. семинара «Практика и перспективы институционального партнерства». Донецк, ДонГТУ, 2001, N 1. С 36 - 47.
6. Peter J. Ashenden. The designer's Guide to VHDL // Morgan Kaufmann, San Francisco California, US 1995г. 688с.
7. Зинченко Ю., Дяченко О., Маркитантов В. и др.. Новые hardware-технологии в ДонГТУ /Материалы межд. научн.-техн. конф. «Новые информационные технологии в САПР и АСУ». - Киев: Украинский дом экономических и научн.-техн. знаний. – 2001. С 12 – 19.
8. Zinchenko Y. E. computer-aided design and hardware description languages / Сб. тр. ДонГТУ. Сер.: Проблемы моделирования и автоматизации проектирования динамических систем, вып. 10. – Донецк: ДонГТУ, 1999.- С. 210-216.

УДК 004.3

АППАРАТНАЯ ОБРАБОТКА И РАСПОЗНОВАНИЕ ЗВУКОВЫХ СИГНАЛОВ НА БАЗЕ FPGA

Азаров А. Б., Константинов В. С., Зинченко Ю. Е., Зинченко Т. А.

Донецкий национальный технический университет, кафедра компьютерной инженерии

E-mail: azarovantony@gmail.com

Аннотация:

Азаров А. Б., Константинов В. С., Зинченко Ю. Е., Зинченко Т. А. Аппаратная обработка и распознавание звуковых сигналов на базе FPGA. Разработана архитектура устройства распознавания импульсных звуковых сигналов на базе FPGA. В качестве основы для построения устройства выбран алгоритм динамической трансформации временной шкалы DTW (dynamic time warping). Для повышения быстродействия предлагается распараллелить процесс распознавания, что вместе с аппаратной реализацией на FPGA позволяет реализовывать эту задачу в реальном масштабе времени. Представлена внутренняя структура FPGA устройства распознавания, построенная по принципу жесткой логики с распределенными связями, приводится алгоритм функционирования устройства.

Annotation:

Azarov A. B., Konstantinov V. S., Zinchenko Yu. E., Zinchenko T. A. Hardware processing and recognition of sound signals based on FPGA. The architecture of the device for recognizing pulsed sound signals based on FPGA has been developed. The algorithm for dynamic transformation of the time scale DTW (dynamic time warping) was chosen as the basis for building the device. To improve performance, it is proposed to parallelize the recognition process, which, together with the hardware implementation on the FPGA, allows realizing this task in real time. Introduced the internal structure of the FPGA recognition device, built on the principle of hard logic with distributed connections, provides an algorithm for the operation of the device.

Введение

Технологии распознавание образов находят все более широкое применение, так как они на практике позволяют решать сложные технические задачи в самых разнообразных сферах человеческой деятельности, там, где другие технологии либо принципиально не в состоянии справиться с задачей либо их применение чрезвычайно сложно. Одной из разновидностей таких задач является распознавание звука. Наибольшее распространение в этой области получила задача распознавания человеческой речи [1-3], другие типы звуков исследованы достаточно слабо. К слабо исследованным типам звуков можно отнести так называемые «одиноким импульсным звукам», которые характеризуются скоротечностью, высокими скоростями нарастания и убывания фронтов импульсов и широким спектром сигналов, на которые можно разложить импульсы звуков. Целью данной статьи является разработка алгоритма и архитектуры системы распознавания импульсных звуковых сигналов на базе программируемых логических схем (ПЛИС, FPGA).

Архитектура устройства

ВКС распознавания звуковых сигналов (РЗС) должна выполнять следующие функции:

- запись звуков;
- выявление всплесков звуковой активности;
- сохранение звуковых файлов пригодных для анализа;
- распознавание звуковых сигналов;

- передача результатов анализа.

Учитывая особенности устройства можно выделить следующие компоненты, необходимые для его автоматической работы:

- ОЗУ для хранения промежуточных вычислений;
- ПЗУ для хранения исходных сигналов и результатов анализа;
- блок обработки данных;
- блок записи сигналов;
- интерфейс отладки;
- блок передачи данных;
- блок управления для координации действий устройства.

Структурная схема устройства представлена на рисунке 1.

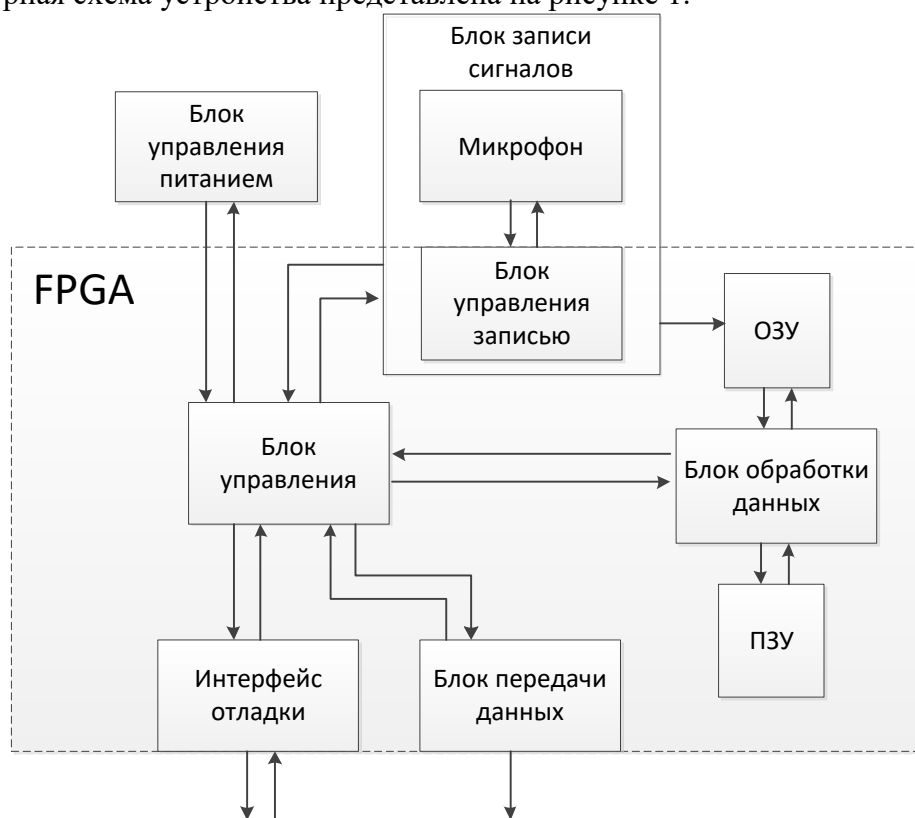


Рисунок 1 – Структурная схема устройства РЗС

Современные FPGA комплексы обладают встроенной памятью, нет необходимости в использовании внешних носителей, однако для удобства пользования в качестве ПЗУ можно использовать SD карту либо USB накопитель, для возможности извлечения данных их анализа в другом месте без транспортировки устройства, а так же возможности дополнения и замены эталонов в случае необходимости.

Для связи с внешними устройствами на плате присутствует несколько интерфейсов. Первый интерфейс используется для загрузки программы в плату и дальнейшей его отладки. Для этого используется USB интерфейс. После обработки сигнала блоком обработки данных результаты должны быть переданы на сервер. С учетом того что устройство может находиться на большом расстоянии от сервера, а также необходимый уровень защищённости, были выбраны RS-485. в качестве беспроводного аналога рассматривались мобильная и радио связь, однако в связи с тем что сотовая связь ряд ограничений, была выбрана радио связь.

Для обеспечения автономной работы в устройстве присутствует цепь питания для возможности использования при отсутствии непосредственного подключения к сети электропитания.

Блок записи звуковых сигналов состоит из двух частей: устройство звукозаписи (микрофон) и блок работы со звукозаписывающим устройством в FPGA комплексе.

Алгоритм работы устройства

Алгоритм работы устройства РЗС можно разбить на последовательность из четырех отдельных шагов.

На первом этапе происходит запись звуковых сигналов при помощи микрофона. В случае обнаружения всплесков звуковой активности происходит запись звука в файл в формате *.wav. Файл хранит только момент всплеска и хранится в отдельной папке на SD-карте. К звуковым файлам, участвующим в анализе предъявляются следующие требования: длительность не более 5 секунд и размер не более 35кВ. При не соблюдении данных требований возможно искажение результатов исследования.

На втором этапе записанные файлы поочередно анализируются. Для анализа файла содержимое звукового файла преобразовывается из дискретного звукового сигнала в вектор значений пригодный для анализа. Для преобразования звуковых сигналов в вектора можно использовать алгоритм быстрого преобразования Фурье. Также при обработке вектора нормализуются для более удобной обработки. Полученные вектора будут сравниваться с уже имеющимися векторами (эталоны), для определения наибольшего сходства. Для сравнения векторов используется алгоритм динамической трансформации временной шкалы. В ходе сравнения каждый звуковой сигнал поочередно сравнивается с каждым из эталонов. Результатом анализа файла будет массив «расстояний» отображающий степень сходства звукового сигнала с каждым их эталонов [4,5].

Перед запуском устройства через интерфейс отладки в него погружается файл эталонов, хранящий вектора соответствующие звуковым сигналам признанные эталонными и предназначенные для определения природы происхождения сигнала. Такой способ загрузки был предусмотрен для удобной смены эталонов в ходе исследования.

Заключительный этап состоит в отправке результата на сервер по завершению анализа звукового сигнала. Данные на сервер предполагается отправлять в общепринятом формате JSON (JavaScript Object Notation) для удобной обработки данных на стороне сервера. Для связи с сервером может использоваться несколько подходов, это LoRaWAN и мобильная передача данных, стандартов - 2G, 3G и 4G. В случае с LoRaWAN данные передаются небольшими пакетами, не более 255 байт, поэтому данные для отправки необходимо делить на блоки, размер которых составляет 255 байт. Интерфейс для отправки данных принимает на вход строку, что очень удобно с форматом JSON.

При обнаружении ошибок в работе устройства или некорректной работы алгоритма, звуковые файлы могут быть извлечены и проанализированы на другом устройстве либо компьютеры.

Ускорение алгоритма РЗС

Для распознавания звуковых сигналов на данный момент используется алгоритм DTW. При использовании данного алгоритма звуковой сигнал поочередно сравнивается с набором эталонов. При таком подходе наблюдается алгоритмическая зависимость времени анализа от количества эталонов. При количестве эталонов равным N, время выполнения алгоритма T будет равно $N*t$, где t – среднее время сравнения с эталоном.

Для устранения данного недостатка можно распараллелить вычисления. Для реализации такого подхода отлично подходит FPGA комплекс. Структура FPGA комплекса

позволяет загрузить в плату М блоков для анализа сигналов. Структура блока РЗС представлена на рисунке 2

Каждый блок проводит сравнение анализируемого сигнала с одним эталоном, при этом все элементы работают параллельно. Следовательно, в частных случаях при N кратном М полное время анализа Т будет равно $N \cdot t / M$.

При других значениях N, время работы будет вычисляться по общей формуле (1)

$$T = \lceil N/M \rceil \cdot t. \quad (1)$$

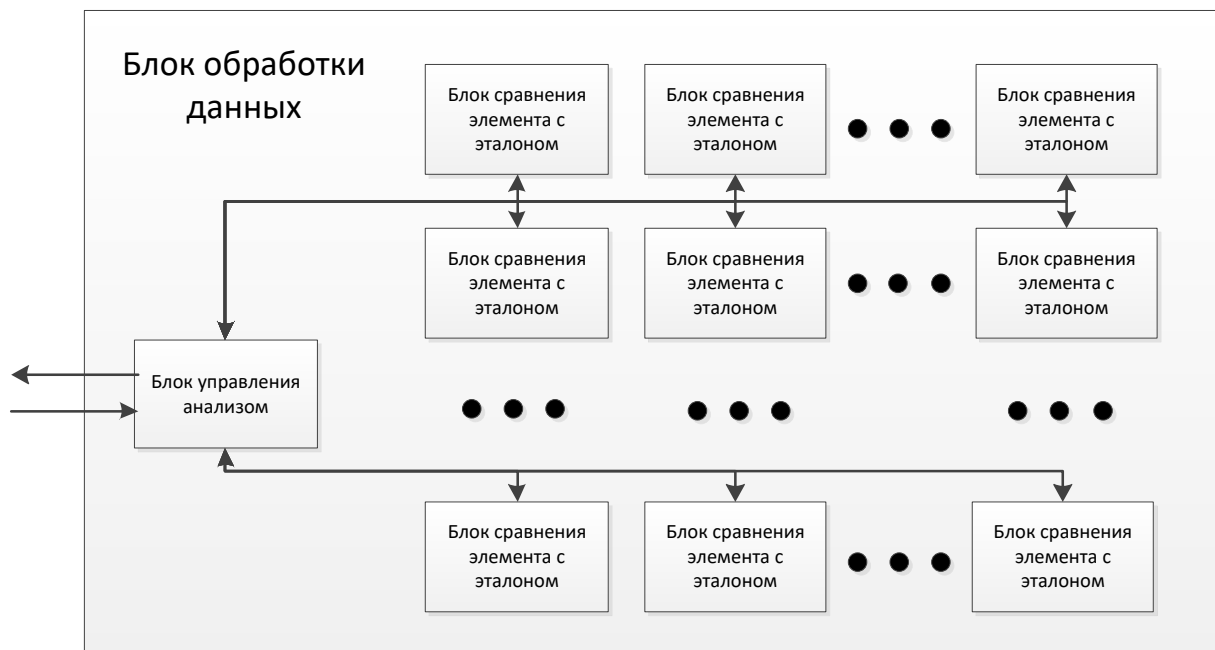


Рисунок 2 – Блок анализа звуковых сигналов

При распараллеливании работы получает уменьшение времени обработки примерно в М раз. Количество блоков М ограничено размером платы.

Выводы

В данной статье была рассмотрена структура устройства для распознавания звуковых сигналов, также был проведен анализ алгоритма работы устройства и предложен. В результате анализа были предложены методы по увеличению производительности устройства путем использования параллельных вычислений. В дальнейшем планируется еще одна попытка усовершенствования алгоритма путем распараллеливания вычислений в алгоритме сравнения звукового сигнала с одним эталоном. Результаты данной работы будут в дальнейшем использованы при подготовке магистерских диссертаций.

Литература

1. Бабаринов С.Л., О распознавании речи [электронный ресурс] / С.Л. Бабаринов, М.А. Будникова – Электронная библиотека – Режим доступа: URL: <https://cyberleninka.ru/article/v/o-raspoznanii-rechi>
2. Грачев А.М., Лингвистические подходы к автоматическому распознаванию речи [электронный ресурс] / А.М. Грачев – Электронная библиотека – Режим доступа: URL: <https://cyberleninka.ru/article/v/lingvisticheskie-podhody-k-avtomaticheskomu-raspoznaniyu-rechi>

3. Шелепов В.Ю. Сегментация и дифонное распознавание речи: [Лекции]/ В.Ю. Шелепов, А.В. Ниценко. – Донецк : ГУ «ИПИИ», 2015. – 232 с.
4. Намиот, Д.Е. Базы данных временных рядов в системах «Интернета вещей» [Текст] / Д.Е. Намиот. – Litres, 2017. – 80 с.
5. Алгоритм динамической трансформации временной шкалы. [электронный ресурс].
Режим доступа: URL: https://ru.wikipedia.org/wiki/Алгоритм_динамической_трансформации_временной_шкалы

УДК 00.004.45

ИССЛЕДОВАНИЕ ДАННЫХ ЖЁСТКОГО ДИСКА С ИСПОЛЬЗОВАНИЕМ ИНСТРУМЕНТОВ UEFI

Чередникова О. Ю., Якубов Я. А.

Донецкий национальный технический университет

Кафедра компьютерной инженерии

E-mail: alexgdi@outlook.com

Аннотация:

Чередникова О.Ю., Якубов Я.А. Исследование данных жёсткого диска с использованием инструментов UEFI. Продемонстрированы возможности UEFI для разработки системных утилит. Рассмотрена разработка приложения на платформе UEFI на примере системной утилиты для исследования данных физического диска.

Annotation:

Cherednikova O. Y., Yakubov Y.A. Research hard drive data using UEFI tools. Demonstrated UEFI features for developing system utilities. The development of an application on the UEFI platform is considered on the example of a system utility for examining physical disk data.

Общая постановка проблемы

На сегодняшний день системные утилиты выполняют различные специализированные задачи, связанные с работой оборудования или операционной системы. Системные утилиты позволяют реализовать возможности, зачастую недоступные без их применения, либо расширяют эти возможности.

Существует большое количество утилит, разработанных для исследования информации физических дисков. Они выполняют такие задачи:

- Определение схемы разметки диска
- Получение таблицы разделов
- Исследование файловых систем на разделах диска

Однако большинство утилит не рассчитаны на запуск вне операционной системы, поэтому актуальной является задача разработки утилиты, которая могла бы определять схему разметки диска без загрузки ОС.

Типы таблиц разделов

Для анализа данных физического диска необходимо знать размер и расположение логических разделов диска, эта информация храниться в таблице разделов. Существует два типа таблиц разделов физических дисков: традиционная и GPT (GUID-таблица разделов).

Традиционная таблица разделов является частью схемы разметки MBR (Master Boot Record, главная загрузочная запись), которая расположена в нулевом секторе физического диска. Размер структуры MBR равен 512 байт. Она использовалась начиная с 1983 года в операционной системе IBM PC DOS 2.0 и на данный момент считается устаревшей. MBR непосредственно участвует в процессе запуска компьютера после окончания процедуры POST (Power-on self-test) и выбора загрузочного устройства. Базовая система ввода-вывода (BIOS) загружает код первичного загрузчика, который располагается в MBR и передаёт ему управление. Первичный загрузчик ищет раздел, на котором находится операционная система и передаёт управление вторичному загрузчику, который расположен на этом разделе и далее загружает операционную систему. Ранее структура MBR использовала CHS (Cylinder, Head, Sector) адресацию, которая отражала реальную геометрию диска, но она позволяла использовать диск объёмом не более 8 гигабайт. С быстрым рост объёма накопителей CHS-

адресация была заменена LBA (Logical Block Addressing) адресацией. Смысл LBA-адресации состоит в том, что весь объём диска разделён на логические блоки. Использование LBA-адресации позволяет использовать раздел, размер которого не превышает 2.2 терабайта. Классическая структура MBR представлена на рисунке 1 [1].

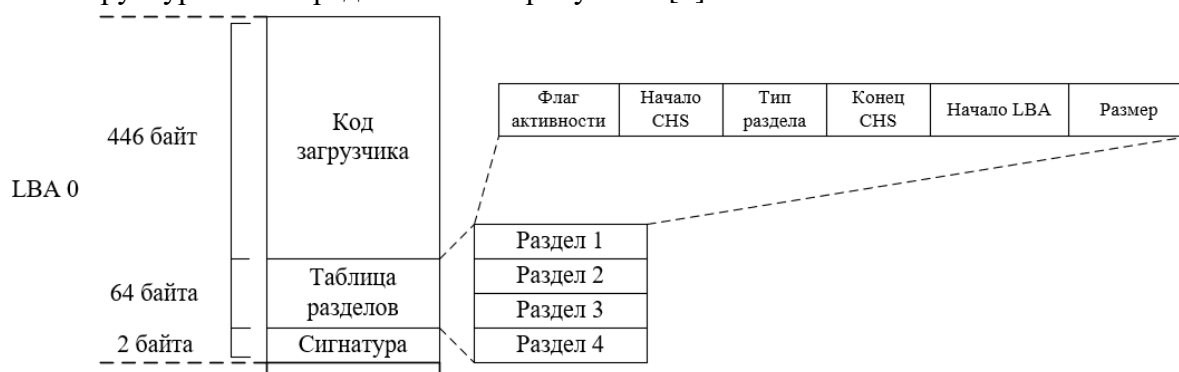


Рисунок 1 - Классическая структура MBR

В 1998 году компания Intel разработала спецификацию EFI (Extensible Firmware Interface) как замену BIOS. В 2005 году EFI была переименована в UEFI (Unified Extensible Firmware Interface). В спецификацию UEFI входит GUID-таблица разделов, которая была разработана для преодоления ограничений, накладываемых традиционной таблицей разделов физического диска, не позволяющей создавать разделы больше 2.2 терабайт. В GPT полностью отказались от адресов в формате CHS, которые ранее использовала главная загрузочная запись. Диски, которые используют GPT, могут содержать защитную MBR, которая используется, если компьютер не поддерживает спецификацию UEFI и защищают GPT от повреждения программами, не знающими о существовании UEFI [1]. Структура GPT представлена на рисунке 2.

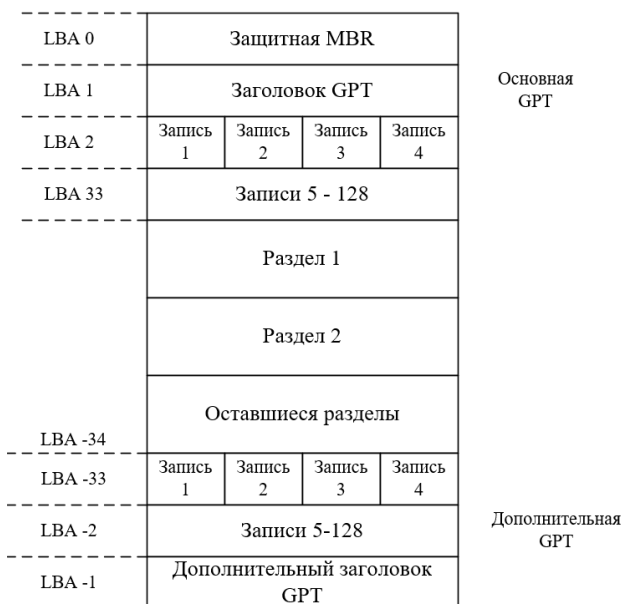


Рисунок 2 - Структура GPT

Возможности UEFI для разработки системных утилит

Основное предназначение UEFI - замена устаревшей технологии BIOS и преодоление связанных с ней ограничений. UEFI является интерфейсом между операционной системой и микропрограммами, управляющими низкоуровневыми функциями оборудования. Его основными функциями являются инициализация и первичное тестирование оборудования при включении системы и передача управление загрузчику ОС. Разработкой спецификации занимается UEFI Forum, в состав которого входят представители различных компаний и

вендоров. UEFI имеет модульную архитектуру, где каждый модуль представляет собой драйвер или приложение, которое реализует дополнительные возможности или поддержку оборудования. Приложения UEFI выполняются непосредственно перед загрузкой операционной системы. Приложения разрабатываются на языке высокого уровня, могут иметь графический интерфейс, который упрощает их использование. Возможность разработки UEFI приложений и драйверов на языке виртуальной машины EFI Byte Code (EBC) позволяет обеспечить кроссплатформенность программного обеспечения и минимизировать затраты на портирование программного кода под различные процессорные архитектуры [2].

Спецификация UEFI имеет две открытые реализации: TianoCore, разработанную Intel, и gnu-efi, разрабатываемую сообществом. Обе реализации опубликованы под лицензией BSD. Реализация TianoCore является более популярной, чем gnu-efi и предоставляет многофункциональное окружение и среду EDK II, которая может использоваться для разработки микропрограмм для UEFI и PI (Platform Initialization Specification) спецификаций. EDK II снабжена примерами и документацией, которые упрощают и ускоряют разработку приложений.

Разработка утилиты UEFI для исследования данных жёсткого диска

Для разработки утилиты была выбрана реализация TianoCore, имеющая большое количество примеров, документации и реализацию стандартной библиотеки языка C. Приложения могут быть драйверами UEFI, которые могут работать резидентно, и приложениями UEFI. Утилита разработана как приложение, так как ей не требуется резидентный режим работы.

UEFI описывает программный интерфейс, который позволяет абстрагироваться от конкретной аппаратной платформы. Аппаратная платформа включает в себя процессор, чипсет и другие компоненты. Основными концепциями UEFI являются:

- Объекты, управляемые UEFI-микропрограммой и использующиеся для управления состоянием системы
- Системная таблица UEFI – основная структура данных, содержащая таблицы данных и вызовы функций для взаимодействия с системой
- База дескрипторов и протоколов – набор предоставляющие вызываемых интерфейсов
- Образы UEFI – формат исполняемых файлов
- События – позволяют приложениям обмениваться сигналами
- Пути к устройствам – структуры данных, которые описывают аппаратное расположение устройств, таких как шина, раздел или имя образа UEFI на диске.

Системная таблица UEFI является самой важной структурой в UEFI. Указатель на структуру передаётся в каждое приложение как параметр в его точку входа. Из этой структуры образ UEFI может получить доступ к конфигурации системы и набору служб: UEFI Boot Services, UEFI Runtime Services и службам протоколов.

База данных дескрипторов и протоколов содержит в себе дескрипторы, идентифицируемые по GUID (Globally Unique Identifier) – уникальному 128-битному идентификатору. Протокол UEFI – блок указателей функций и структур данных, которые определены спецификацией UEFI. Каждый протокол также имеет GUID, который используется функцией HandleProtocol, для получения дескриптора этого протокола. Протокол может включать в себя структуру данных, содержащую поля данных или сервисы. Во время инициализации UEFI, приложения и драйверы присваивают протоколы дескрипторам. Дескрипторы являются глобальными и могут быть использованы любым образом UEFI. Каждый дескриптор идентифицируется уникальным номером дескриптора. Каждая запись в базе дескрипторов представляет собой набор из одного или нескольких

протоколов. Тип протоколов определяется GUID, присоединяется к дескриптору и определяет тип дескриптора. Дескриптор может указывать на такие компоненты, как образ UEFI, устройство, службы UEFI.

Для получения доступа к физическим дискам был использован EFI_BLOCK_IO_PROTOCOL, позволяющий использовать физический диск как блочное устройство. Для получения дескриптора протокола необходимо вызвать функцию LocateHandlerBuffer, указав GUID запрашиваемого протокола. Поле глобальной переменной gBS, которая является указателем на UEFI Boot Services, содержит указатель на функцию LocateHandlerBuffer. Данная функция при вызове вернет все дескрипторы, которые указывают на устройства, поддерживающие передаваемый ей протокол. Далее, вызвав функцию HandleProtocol и передав ей дескриптор, мы получим объект протокола блочного устройства [3].

Блочными устройствами могут являться как физические диски, так и логические. Объект протокола содержат указатели на такие функции как ReadBlocks, WriteBlocks и другие. Эти функции позволяют записывать и считывать данные на диск поблочно.

С использованием этих функций производится анализ содержимого физического диска, анализ структур MBR и GPT, получение таблицы разделов. Интерфейс программы представлен на рисунке 3.

```
Disk list:
| Disk 0 | DiskSize: 99 MB | PT_Type: MBR | BlockSize: 512 bytes
Partition 1. Size: 50 MB Type: FAT32
Partition 2. Size: 47 MB Type: NTFS

| Disk 1 | DiskSize: 199 MB | PT_Type: GPT | BlockSize: 512 bytes
Partition 1. Size: 15 MB Type: Microsoft reserved partition
Partition 2. Size: 49 MB Type: Basic data partition
Partition 3. Size: 49 MB Type: Basic data partition
Partition 4. Size: 81 MB Type: Basic data partition

| Disk 2 | DiskSize: 15007 MB | PT_Type: GPT | BlockSize: 512 bytes
Partition 1. Size: 15005 MB Type: Basic data partition
```

Рисунок 3 - Интерфейс программы

Утилита позволяет определить таблицу разделов для физических дисков MBR и GPT, определить объём физического диска, объём и тип каждого из его разделов.

Выводы

В результате были продемонстрированы возможности UEFI по разработке системной утилиты для исследования данных физического диска. Разработанная утилита может выполняться на компьютере, оснащённом микропрограммой, разработанной в соответствии со спецификацией UEFI, без наличия на нём установленной операционной системы.

Литература

1. Кэрриэ Б. Криминалистический анализ файловых систем [Текст]. – СПб.: Питер, 2007. – 480 с.: ил.
2. Зиммер, В. По ту сторону BIOS. Разработка с Unified Extensible Firmware Interface / Зиммер, В, Ротман М., Марисетти С. – 3-е изд., 2017. – 325 с.
3. Unified Extensible Firmware Interface (UEFI) Specification. Version 2.7 Errata A – Unified EFI Forum, 2017. – 2575 с.

УДК 004.77, 004.056.523

МЕТОДЫ РАЗГРАНИЧЕНИЯ ПРАВ ДОСТУПА К УДАЛЕННЫМ РЕСУРСАМ

Чередникова О. Ю., Зимцев П. П.

Донецкий национальный технический университет,
г. Донецк, Донецкая народная республика

Аннотация

Чередникова О. Ю., Зимцев П. П. Методы разграничения прав доступа к удаленным ресурсам. В статье исследованы и проанализированы классические модели разграничения доступа к ресурсам, рассмотрены их недостатки. Проведен анализ принципов работы системы разграничения доступов на основании основных базовых составляющих элементов модели Role Based Access Control (RBAC). Описана модель ролевого разграничения и приведена ее реализация посредством языка программирования - Go!.

Ключевые слова: разграничение доступа, модель, RBAC, программирование, права доступа, правила.

Abstract

Cherednikova Olga, Zimtsev Petr. Methods of differentiation of access rights to remote resources. The article studies and analyzes classical models of access control to resources, considers their shortcomings. The analysis of the principles of the system of access control on the basis of the basic components of the model elements Role Based Access Control (RBAC). The disadvantages of the use of classical models of access control to resources are considered. The model of role differentiation is described and its implementation is given through the programming language - Go!.

Keywords: access control, model, RBAC, programming, access rights, rules.

Введение

Характерной особенностью современных информационных систем является необходимость сочетания богатого информационного наполнения с передовыми технологическими разработками. Это обусловлено тем, что наряду с новыми данными и документами большое значение может иметь наличие особых методов обработки информации, доступа к хранилищу документов, структурных и семантических связей с остальными компонентами информационной системы.

Одним из необходимых условий функционирования такой системы является задача разграничения доступа к ресурсам системы, т.е. поддержание требуемого уровня конфиденциальности, целостности и доступности данных. Отметим, что под ресурсами системы понимаются не только сама информация, но и свободное место на дисках, процессорах и т.д.

Классические модели разграничения доступа к ресурсам

В самом общем виде доступ к ресурсам можно представить в виде матрицы, задающей соотношение субъектов и объектов. К субъектам относятся авторизированные пользователи системы, причем один физический пользователь может соответствовать нескольким субъектам в контексте разграничения доступа. Объектами являются всевозможные ресурсы системы: документы, базы данных, рубрикаторы, файлы, внешние устройства и т.д. К объектам могут относиться и сложные структуры, описывающие пользователей системы, их группы, организации и т.п. В матрице доступа каждому субъекту

соответствует строка, а объекту - столбец. На их пересечении указываются права на действия, которые субъект имеет право производить над объектом.

Однако прямой путь реализации полной матрицы доступа в реальных системах неприемлем: в этих системах функционируют сотни и тысячи субъектов, выполняя различные действия над десятками и сотнями тысяч объектов. Полное число элементов типичной матрицы доступа будет составлять миллиарды, что совершенно неприемлемо как с точки зрения производительности, так и с точки зрения поддержания ее целостности и актуальности.

Одним из путей решения указанной проблемы является реализация элементов матрицы доступа по мере надобности на основе атрибутов, приписываемых объектам и субъектам по правилам, определяемым так называемыми “политиками доступа”.

Недостатки применения классических моделей разграничения доступа в информационных системах

Использование классических средств разграничения доступа, реализованных в рамках операционной системы и СУБД, оказывается совершенно недостаточным для эффективного и надежного функционирования информационных систем (ИС). Приведем только два фактора, которые объясняют причины указанных явлений.

Во-первых, информационные системы, с точки зрения операционной системы, являются однопользовательскими, т.е. все процессы работают от лица одного пользователя, как правило, непривилегированного, пользователя, который ни в коем случае не должен являться владельцем информационного хранилища[1]. Этим достигается определенный уровень безопасности со стороны операционной системы, например гарантированное использование места на дисках, загрузка процессора. Хранилище информации зачастую является гетерогенным - комбинацией файлового хранилища для документов и базы данных для хранения метаданных системы, и распространяется на разных серверах. Владелец хранилища является специальный непривилегированный пользователь, который имеет право на модификацию.

Еще одним фактором, затрудняющим использование классических схем контроля доступа, является наличие большого количества объектов в ИС, требующих ограничение доступа[1].

Например, для ведения каталога ресурсов требуется контролировать действия как над отдельными структурными частями каталога, так и над возможностью доступа к ресурсам, приписанным к разным веткам каталога. Как правило, работа над каталогом ведется редакторской группой, при этом каждый редактор обладает разной квалификацией, компетенцией и уровнем ответственности. Ограничение доступа здесь преследует несколько целей:

- конфиденциальность, редактор может работать только со своей рубрикой;
- целостность, структура каталога может быть изменена только очень компетентными редакторами;

Другой пример - регистрация автора в системе.

Описание модели ролевого разграничения

Пользователи и объекты здесь понимаются в обычном смысле, т.е. как авторизованные пользователи системы и как множество физических и логических объектов, доступ к которым регулируется с помощью модели Role Based Access Control (RBAC).

Привилегии - это минимальное возможное атомарное действие пользователя, которое требует того или иного разграничения доступа к этому действию. В принципе возможно объединение нескольких атомарных действий в одну привилегию, но только при условии, что во всех возможных ситуациях будет требоваться именно такая комбинация

простейших действий. Объекты обладают атрибутами, которые могут фигурировать в правилах, составляющих ту или иную роль[2].

Роль - это набор правил (прав), определяющих, какими привилегиями и над какими объектами будет обладать пользователь, которому присваивается данная роль. Правила, составляющие роль, могут быть разрешающими или запрещающими.

Права (правила) - составная часть роли, определяющая привилегию, подмножество объектов, обладающих данной привилегией, и разрешение или запрет на выполнение данного действия.

Сессия - подмножество активных в течении некоторого интервала времени ролей данного пользователя. Установление сессии RBAC возможно только после регистрации пользователя в системе. Активация каких либо ролей вне сессии невозможна. Окончание сессии может производиться явным образом самим пользователем или системой RBAC по окончании определенного интервала времени отсутствия активности и т.д. Одновременно могут выполняться несколько сессий для одного и того же пользователя[2].

Функционирование системы RBAC иллюстрирует схема, показанная на рис.1.

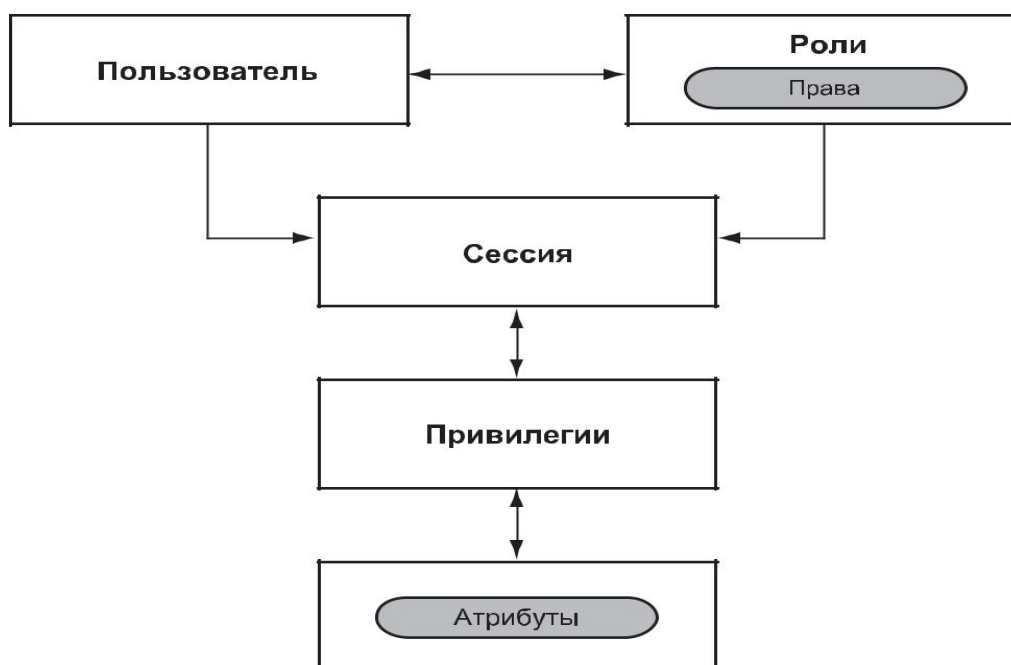


Рисунок 1 - Схема функционирования системы RBAC

Разработчик приложений, выполняющий различные операции над объектами системы, совместно с администратором RBAC и конструкторами ролей RBAC составляют список привилегий.

Конструкторами ролей разрабатывается библиотека ролей данной системы.

Диспетчерами прав пользователей каждому пользователю системы статическим образом присваивается набор ролей.

Сразу после авторизации пользователя в системе для него создается сессия работы с RBAC. Новые сессии создаются явным способом по команде пользователя или неявным способом по окончании предыдущей сессии, если работа пользователя с системой продолжается[4].

При попытке пользователями выполнить какое-либо действие над объектом, контролируемым RBAC, подсистема RBAC выполняет следующие действия:

- проверяет, является ли данное действие допустимым для данного пользователя в рамках текущей сессии RBAC с учетом атрибутов объекта на момент выполнения обращения;
- если действие входит в одну из ролей, присвоенных данному пользователю, - оно выполняется;
- в противном случае пользователь получает сообщение о недопустимости данного действия над объектом.

Легко видеть, что в рамках системы ролей можно воспроизвести как мандатную, так и дискретную политику разграничения доступа к ресурсам.

Стандарт, регламентирующий терминологию, принципы и функциональность ролевой системы доступа, разработан Национальным институтом стандартов и технологий США[4].

Программная реализация с использованием языка программирования Go!

Рассмотрим программную реализацию модели RBAC. Система распределения доступом предполагает разрешение или запрет доступа для просмотра или редактирования данных к ресурсам “Монитора”. Под определением “Монитор” понимается совокупность проверок на доступность подключения какого-либо ресурса.

Например, существует ресурс, состояние которого необходимо анализировать. Ресурс имеет url: “<http://example.com>” и необходимо проверять на доступность по http, ftp, ssh, ssl - сертификат. При получении информации, функция AccessToMonitor проверяет права пользователя, а именно:

- роль, при доступе для просмотра нет необходимости проверки, в ином случае проверяется принадлежность пользователя к ролям "owner" или "admin";
- проверка на вид доступа, имеет пользователь доступ к только одному данному монитору, к группе мониторов, в которую входит запрашиваемый монитор или же пользователь является владельцем или администратором аккаунта, к которому принадлежит запрашиваемый монитор, и имеет доступ ко всем действиям над мониторами[3].

“Монитор” принадлежит только к одному аккаунту. В свою очередь пользователи, обладающие ролями “owner” и ”admin” имеют давать возможность просмотра или редактирования “монитора”.

Ниже представлена реализация проверки доступа пользователя к “монитору” по таким атрибутам как роль и атрибут доступа к “монитору”, “группе мониторов” и “аккаунту”.

```
func (a *Ability) AccessToMonitor(monitor *models.Monitor, val string) bool {
    for _, ac := range a.accesses {
        flag := false
        if val == "view" {
            flag = true
        } else {
            flag = ac.Role == "owner" || ac.Role == "admin"
        }
        switch ac.Kind {
        case "monitor":
            if *ac.MonitorID == monitor.Model.ID && flag {
                return true
            }
        case "group":
            if *ac.GroupID == monitor.GroupID && flag {
                return true
            }
        case "account":
```

```
        if flag {
            return true    }
        }
    }
    return false
}
```

Представленная функция вызывается при попытке обращения к объекту посредством HTTP запроса. В случае провального ответа, когда пользователь не имеет прав на определенные действия, возвращается ошибка доступа. Иначе доступ для дальнейших операций над “монитором” разрешается.

Пример вызова функции:

```
if !a.AccessToMonitor(check.Monitor, "view") {
    ForbiddenError(c, "")
    return nil    }
```

Выводы

Рассмотрена классическая модель разграничения доступов к ресурсам. Проанализирован принцип работы классических моделей разграничения доступа к ресурсам. Описаны и исследованы основные недостатки применения классических моделей. Исследованы основные принципы построения систем разграничения доступов на примере модели разграничения доступов RBAC. Рассмотрена программная реализация распределения доступов в модели RBAC.

Литература

1. A Proposed Standard for Role Based Access Control (PDF) / D.Ferraiolo, R. Sandhu, S. Gavrila et al. // ACM Transactions on Information and System Security. 2001. Vol. 4. N 3. Draft of a consensus standard for RBAC <http://csrc.nist.gov/rbac/rbacSTD-ACM.pdf>
2. RBAC LIST (Laboratory for Information Security Technology, George Mason University) <http://list.gmu.edu/>.
3. Донован А., Керниган Б., Брайан У. Язык программирования Go. : Пер. с англ. - М.: ООО “И.Д. Вильямс”, 2016. - 432 с. : ил. - Парал. тит.англ.
4. Osborn S., Sandlu R., Munawer Q. Configuring Role-Based Access Control to Enforce Mandatory and Discretionary Access Control Policies // ACM Transactions on Information and Systems Security (TISSEC). 2000. Vol. 3. N 2.

УДК 004.7

ВЗАИМОДЕЙСТВИЕ РАЗЛИЧНЫХ SCADA-СИСТЕМ В ЦИФРОВЫХ ПРОМЫШЛЕННЫХ СЕТЯХ

Чередникова О. Ю., Коротач А. С.

Донецкий национальный технический университет
кафедра компьютерной инженерии
E-mail: Akorotach@yandex.ru

Аннотация:

Чередникова О.Ю., Коротач А.С. Взаимодействие различных SCADA-систем в цифровых промышленных сетях. Рассмотрены тенденции развития автоматизации производств с использованием цифровых промышленных сетей и проблема обмена данными между такими сетями. Наибольшее внимание уделяется SCADA-системам и вопросам их согласованной работы между собой и со специализированным программным обеспечением.

Annotation:

Cherednikova O.Yu., Korotach A.S. Interaction of various SCADA-systems in digital industrial networks. Trends in the development of industrial automation using digital industrial networks and the problem of data exchange between such networks are considered. The greatest attention is paid to SCADA-systems and the issues of their coordinated work among themselves and with specialized software.

Общая постановка проблемы

В настоящее время развитие автоматизации производств промышленного типа характеризуется увеличением числа «интеллектуальных» конечных устройств и микроконтроллерных систем для управления и контроля над процессами вычислительных систем. Таким образом, сильно возрастает роль информационных данных, которые собираются на всех уровнях автоматизированных систем управления технологическим процессом (АСУ ТП). Требования все более ужесточаются от потребителей итоговой информации. Также ими затрагиваются проблемы скорости и надежности получения данных.

Для решения этих задач активно используются цифровые промышленные сети (ЦПС). Такие сети являются многоузловыми, а передача данных в них происходит при помощи цифрового обмена. На сегодняшний день на рынке представлено около сотни различных типов ЦПС, применяемых в системах автоматизации [1]. Ведущую роль в АСУ ТП играют также SCADA-системы (программы, обеспечивающие создание программного обеспечения для автоматизации контроля и управления технологическим процессом в режиме реального времени). Основная цель создаваемой с помощью SCADA программы – дать оператору, управляющему технологическим процессом, полную информацию об этом процессе и необходимые средства для воздействия на него. Однако, в настоящее время наблюдается тенденция появления большого количества ЦПС, различные узлы которых используют различные SCADA-системы. Поэтому появляется проблема объединения и использования данных, полученных на основе сбора информации с различных SCADA-систем.

Способы обработки данных в ЦПС

В различных отраслях как производства, так и IT-сферы, имеются классы объектов управления и контроля, в автоматизации наблюдения за которыми основное значение имеет качество управления, а не быстрдействие. Создание систем управления для таких объектов связано с большими временными затратами на выбор и реализацию алгоритма управления.

Стоит заметить, что алгоритмизация данного процесса обычно может выполняться на уровне программируемого логического контроллера или SCADA-системы на базе программного логического контроллера, скриптов, VBA, VSTA или других технологий.

Непосредственно разработка алгоритмов и их первичная проверка выполняется в специализированном программном обеспечении таком как: MATLAB, LabVIEW, Mathematica, Maple и т.д. Прямой перенос алгоритмов на программируемый логический контроллер или в SCADA систему требует больших временных затрат, причиной тому – особенности реализации языков программирования как программного обеспечения, так и средств автоматизации.

Как правило, в состав SCADA-систем входит значительное количество различных модулей как для наблюдения и управления, так и для транспортировки данных между участками сети и передача их между различными сетями для дальнейшей обработки и мониторинга результатов (рис.1).

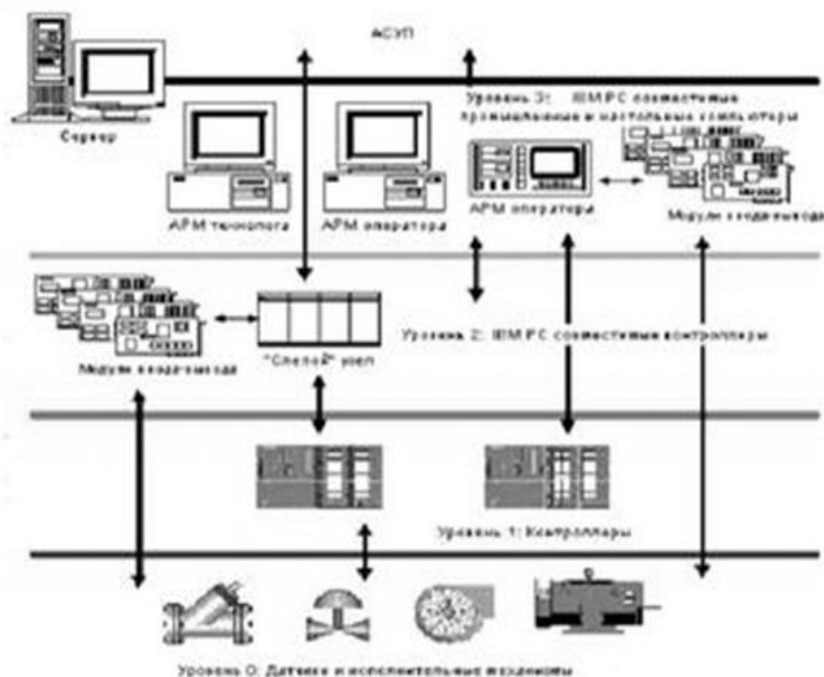


Рисунок 1 - Структура SCADA-системы

Для установки связи с системами высокого уровня используются следующие модули: DEC Slave, DNP3 Slave, ICCP/TASE.2, IEC61870-5-01/104 Slave, Modbus Slave, OPC DA Server, OPC UA Server, SNMP Server, SQL Online.

Значительный вклад в вопрос открытости систем автоматизации был внесен стандартом OPC (OLE for Process Control), что переводится как OLE для управления процессом, обеспечивший наличие широчайшего выбора аппаратного обеспечения, используемого системными интеграторами. Разработчики контроллерного оборудования получили, в свою очередь, от внедрения стандарта OPC-расширение рынков сбыта. Стоит отметить, что данное аппаратное обеспечение совместимо с любыми стандартными SCADA-системами.

Обмен информацией SCADA-систем на базе модуля OPC DA

Рассмотрим один из способов получения данных со SCADA-системы для их дальнейшей обработки и взаимодействия — протокол OPC DA (Data Access). Он обеспечивает обмен данными с устройствами или программными компонентами [3].

Технология OPC DA является клиент-серверной архитектурой. В качестве OPC клиента может выступать любое программное обеспечение, поддерживающее один из интерфейсов: "Custom" или "Automation". OPC-клиент обычно устанавливается на том же компьютере, на котором установлена SCADA-система. OPC клиенты обычно создаются разработчиками SCADA систем.

В качестве OPC сервера может выступать специализированное программное обеспечение (СПО), имеющее COM интерфейс и обеспечивающее доступ к объектам автоматизации и программному обеспечению при помощи интерфейса программирования приложений "Native API". OPC серверы опираются на коммуникационные протоколы представленного оборудования. Компьютер, обеспечивающий связь SCADA системы с оборудованием может иметь OPC сервера с различными спецификациями. Они разработаны на базе различных коммуникационных протоколов (например, AS511, RK512, S-bus, Modbus, DF1 и т.д.), и легко "распознаются" подключенным оборудованием (например, ПЛК).

Коммуникационные протоколы хранятся в специальной базе данных, перед началом работы производится настройка OPC сервера на конфигурацию подключенного оборудования.

Для работы с OPC-клиентом для передачи данных между SCADA-системами мы используем .NET платформу, которая является полностью самостоятельной, однако, накладывает определенные ограничения на работу. Соединение SCADA-систем посредством OPC-канала производится через оболочку RCW (runtime callable wrapper) (рис.2).

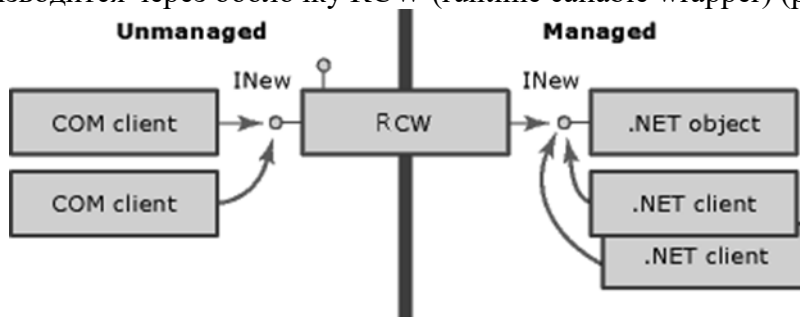


Рисунок 2 - Взаимодействие SCADA-данных на основе СПО с использованием RCW

Основной причиной использования .NET служит портируемость типов данных из COM в .NET с подстройкой их в различного вида СПО. Однако, довольно сложным становится режим обхвата контроля за выполнением операций. В случае .NET функции контроля выполнения метода исполняет механизм исключений, т.е. необходимо использовать try catch .

Среда выполнения создает одну вызываемую оболочку COM для управляемого объекта независимо от числа клиентов COM, которым требуются его службы. Несколько клиентов COM могут содержать ссылку на вызываемую оболочку COM, предоставляющую интерфейс INew. Вызываемая оболочка COM, в свою очередь, содержит единственную ссылку на управляемый объект, который реализует интерфейс. Клиенты COM и .NET могут одновременно выполнять запросы к одному и тому же управляемому объекту SCADA-системы.

Было разработано приложение на языке C# обработки данных через OPC-клиент для взаимодействия различных по оборудованию SCADA-систем для согласованной работы и передачи информации на дальнейшую обработку и сортировку.

Для работы с передачей данных, основанной на СПО с использованием технологии .NET и COM нам необходима библиотека OPCEnumLib. Сообщество OPC Foundation была выпущена специализированная утилита OPCEnum. Эта утилита находится в свободном пользовании. Большинство SCADA систем, являясь OPC-клиентами по умолчанию, при установке инсталлируют указанную утилиту. OPCEnum содержит в себе библиотеку типов,

описывающую инкапсулированные интерфейсы. После добавления в проект ссылки на эту библиотеку, появится возможность использования адресного пространства OPCEnumLib.

Также следует следить за специфическими параметрами подключения серверов к СПО и проверять типы данных. Для этого необходимо использовать библиотеку OPCProxy c:\Windows\system32\opcproxy.dll.

Затем нам требуется импортировать сведения о типах данных из нее:

```
c:\Windows\tlbImp.exe c:\Windows\system32\opcproxy.dll /out:c:\SCADA\opcproxy.dll
```

Теперь возможно создать объект списка OPC серверов и получить идентификатор категории OPC DA 2.0:

```
OpcServerList pServerList = new OpcServerList();
```

```
Guid clsidcat = new Guid("{63D5F432-CFE4-11D1-B2C8-0060083BA1FB}");
```

Далее, вызывая метод EnumClassesOfCategories класса OpcServerList, получаем GUID серверов.

Спецификация OPC DA 2.0 и выше регламентирует необязательный интерфейс OPCBrowseServerAddressSpace. Не смотря на то, что спецификация не требует его обязательной реализации, у большинства серверов он реализован, что дает возможность клиенту просмотреть его внутреннее строение.

Также нам постоянно необходимо опрашивать подключенные SCADA-системы на предмет соединения, так как сервера могут быть установлены на достаточно большом расстоянии друг от друга и для переподключения их к общей сети может быть потрачено довольно большое количество времени, не говоря уже о потере данных при ситуации, когда данные замещаются другими и переносятся без сохранения на другую SCADA, которая не сможет принять сигнал. Таким образом, мы можем обеспечить связь SCADA-системами для более качественной и быстрой обработки данных со SCADA-систем различных производителей при помощи СПО. Также это дает возможность делать бэкапы данных путем подключения через другую SCADA-систему и добывания данных даже при не критических повреждениях данных.

Заключение

Развитие цифровых промышленных сетей и способов для их применения, модернизации и повышения скорости передачи данных не стоят на месте. В данный момент многие промышленные объекты уже оборудованы данным видом сетей для улучшения качества собираемых данных об объектах и уменьшения ошибок на производстве [4]. Поэтому вопрос оптимизации взаимодействия и обеспечения взаимосвязи и обмена информацией становится крайне важным аспектом работы SCADA-систем и программ, связанных с ними.

Литература

1. Максимова Е. А., Грицюк С. Н. Использование SCADA-технологий в современных автоматизированных системах управления // Молодой ученый. — 2015. — №22.5. — С. 45-48. — Режим доступа: <https://moluch.ru/archive/102/23624>
2. Майкл Сатран – About Dynamic Data Exchange. Электронный ресурс. Режим доступа: <https://docs.microsoft.com/en-us/windows/desktop/dataxchg/about-dynamic-data-exchange>
3. Игорь Евлуд - Шпаргалка по OPC DA 2 в .NET. Электронный ресурс. Режим доступа: <https://habr.com/ru/post/132780/>
4. Алексей Волков – SCADA: В поисках идеала. Электронный ресурс. Режим доступа: <https://habr.com/ru/post/319318/> В последние годы вновь увеличился спрос на усилители класса D, благодаря их высокому КПД и достижениям в полупроводниковой технологии.

УДК 004.45

РАЗРАБОТКА УТИЛИТЫ АНАЛИЗА СХЕМЫ РАЗМЕТКИ ЖЕСТКОГО ДИСКА

Чередникова О. Ю., Красников Р. С.

Донецкий национальный технический университет
кафедра компьютерной инженерии

E-mail: krasnikov23ru@gmail.com

Аннотация:

Чередникова О.Ю., Красников Р.С. Разработка утилиты анализа схемы разметки жесткого диска. В статье описана разработка утилиты анализа схемы разметки жесткого диска. Обоснована актуальность разработанного программного обеспечения для изучения структуры информации о логических дисках. Проанализированы утилиты, обладающие похожим функционалом, указаны преимущества разработанного приложения. Приведены анализ схем разметки жесткого диска и алгоритмы просмотра таблицы разделов.

Annotation:

Cherednikova O.Yu., Krasnikov R.S. Development of the utility for analyzing the layout of the hard disk layout. The article describes the development of a utility for analyzing the layout of a hard disk. The relevance of the developed software for studying the structure of information about logical disks is grounded. The utilities with similar functionality are analyzed, the advantages of the developed application are indicated. An analysis of the layout of the hard disk and partition table viewing algorithms.

Общая постановка проблемы

Одной из актуальных задач при изучении курса системного программного обеспечения является задача по исследованию схемы разметки жесткого диска и устройства таблицы разделов. Эта тема является основополагающей при подготовке молодых специалистов в области операционных систем. Однако, при попытке на практике исследовать структуру таблицы разделов, можно столкнуться с проблемой, что существующее программное обеспечение не позволяет в полной мере проанализировать данную таблицу.

Цель работы

Целью работы является разработка программного обеспечения, позволяющего в полной мере проанализировать структуру и содержимое таблицы разделов жесткого диска и отображение этой информации в удобном для пользователя виде. Основной задачей является разработка и реализация алгоритма чтения и анализа данных из таблиц разделов, построенных по принципу Master Boot Record (MBR) [1] и GUID Partition Table (GPT).

Исследование структуры таблицы разделов

До недавнего времени структура Master Boot Record использовалась на всех персональных компьютерах. Данная структура размещена на первом секторе жесткого диска и занимает 512 байт[2]. Первые 446 байт содержат код загрузчика. Этот код предназначен для того, чтобы проанализировать таблицу разделов и передать управление загрузочному разделу. Далее идет таблица разделов. Она состоит из 4 записей по 16 байт. В таблице разделов хранится информация о типе раздела и месте его расположения на диске. В случае, если на диске размещено больше разделов, чем может содержать таблица разделов, то

появляется необходимость в расширении таблицы разделов, путем создания вторичных MBR. Последние 2 байта структуры загрузочной записи занимает сигнатура [3]. На рис. 1 предоставлена структура главной загрузочной записи.



Рис. 1. Структура главной загрузочной записи (MBR)

В связи с тем, что индустрия производства жестких дисков крайне стремительно развивается, постоянно увеличивая их объем, стало понятно, что структура Master Boot Record не позволяет использовать жесткие диски, объем которых превышает 2,2 терабайта. Это объясняется ограничением разрядности полей, хранящих физический адрес раздела – сторону, дорожку и сектор. По этой причине была разработана новая структура GUID Partition Table (GPT) (рис.2) [4].

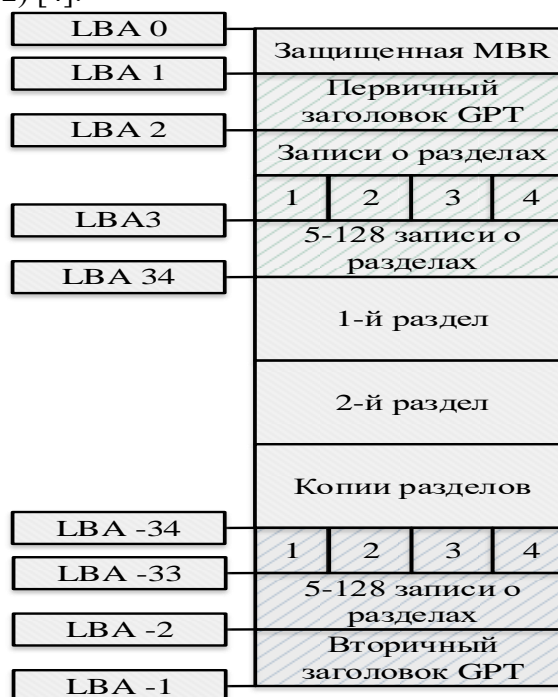


Рис. 2. Структура таблицы разделов GPT

GPT использует адресацию логических блоков (LBA). Первые 512 байт структуры занимает защитная MBR. Это необходимо для предохранения диска, так как множество дисковых

утилит могут не распознать и даже переписать GPT диск. По этой причине в защитной MBR находится одна запись о разделе, который занимает весь GPT диск. Следующий блок содержит оглавление таблицы разделов. Оглавление указывает на те логические блоки, которые могут быть задействованы пользователем. Оно также указывает на число и размер записей данных о разделах. Стандартно, в Microsoft Windows резервируется 128 записей данных о разделах. Таким образом в операционной системе Windows возможно создать до 128 разделов на диске. За заголовком расположены записи данных о разделе. Первые 16 байт каждой записи определяют Globally Unique Identifier (GUID) тип раздела. Далее расположен GUID, уникальный для каждого раздела. Далее записываются данные о начале и конце 64-битных LBA, если они имеются. Остальное место отводится информации об именах и атрибутах разделов. На рисунке 2 представлена структура таблицы разделов GPT.

В настоящее время самой популярной утилитой, позволяющей просмотреть и даже изменить информацию на жестком диске, в частности, схему разметки диска, является утилита DiskEditor. Однако, это приложение не всегда позволяет корректно и доступно для пользователя исследовать структуру вторичных MBR. Поэтому особенностью разработанного программного обеспечения является удобное для пользователя отображение информации обо всех логических дисках жесткого диска.

Разработка приложения анализа разметки жесткого диска

В процессе разработки утилиты были решены следующие задачи:

1) Перебор всех жестких дисков в системе. Для реализации этой задачи была разработана функция ListDisk. В данной функции реализуется цикл, в теле которого, происходит попытка обратиться к жесткому диску при помощи WinAPI функции CreateFile по имени PHYSICALDRIVE, к которому добавляется порядковый номер. В случае, если в системе просмотрены все жесткие диски, функция CreateFile вернет код ошибки, который будет сигнализировать о завершении поиска жестких дисков в системе.

2) Определение схемы разметки диска (MBR или GPT). Это выполняется анализом 450 байта первого сектора жесткого диска. По этому адресу хранится информация о файловой системе первого раздела таблицы MBR. В случае, если данный байт имеет значение 0xED или 0xEE, данный диск имеет разметку GPT. В противном случае диск имеет разметку MBR.

3) Анализ информации диска со схемой разметки MBR. Для анализа дисков со схемой разметки MBR в соответствии с приведенной выше структурой MBR была разработана функция ReadMBR (рис.3). Параметром функции ReadMBR является адрес, с которого нужно читать MBR. После чтения MBR производится цикл прохода по четырем партициям (разделам) Partition Table. Если код типа раздела указывает на расширенный раздел, то производится рекурсивный вызов функции ReadMBR. Значение, хранящееся в партиции по смещению 8 (OTN_ADR на блок-схеме), для не расширенных разделов является адресом начала раздела (адрес BOOT-сектора). Для расширенных разделов это поле содержит адрес вторичной MBR относительно адреса первой вторичной MBR (для первой вторичной MBR это абсолютный адрес).

4) Анализ информации диска со схемой разметки GPT. Для анализа информации о диске с разметкой GPT необходимо прочитать первые 16384 байта. Информация о GPT начинается с 512 байта, с которого начинается заголовок GPT. В нем хранится информация о начале таблицы разделов, о копии таблицы разделов и количестве логических блоков. В отличие от функции для чтения MBR, в функции для чтения GPT нет необходимости рекурсивного вызова функции.

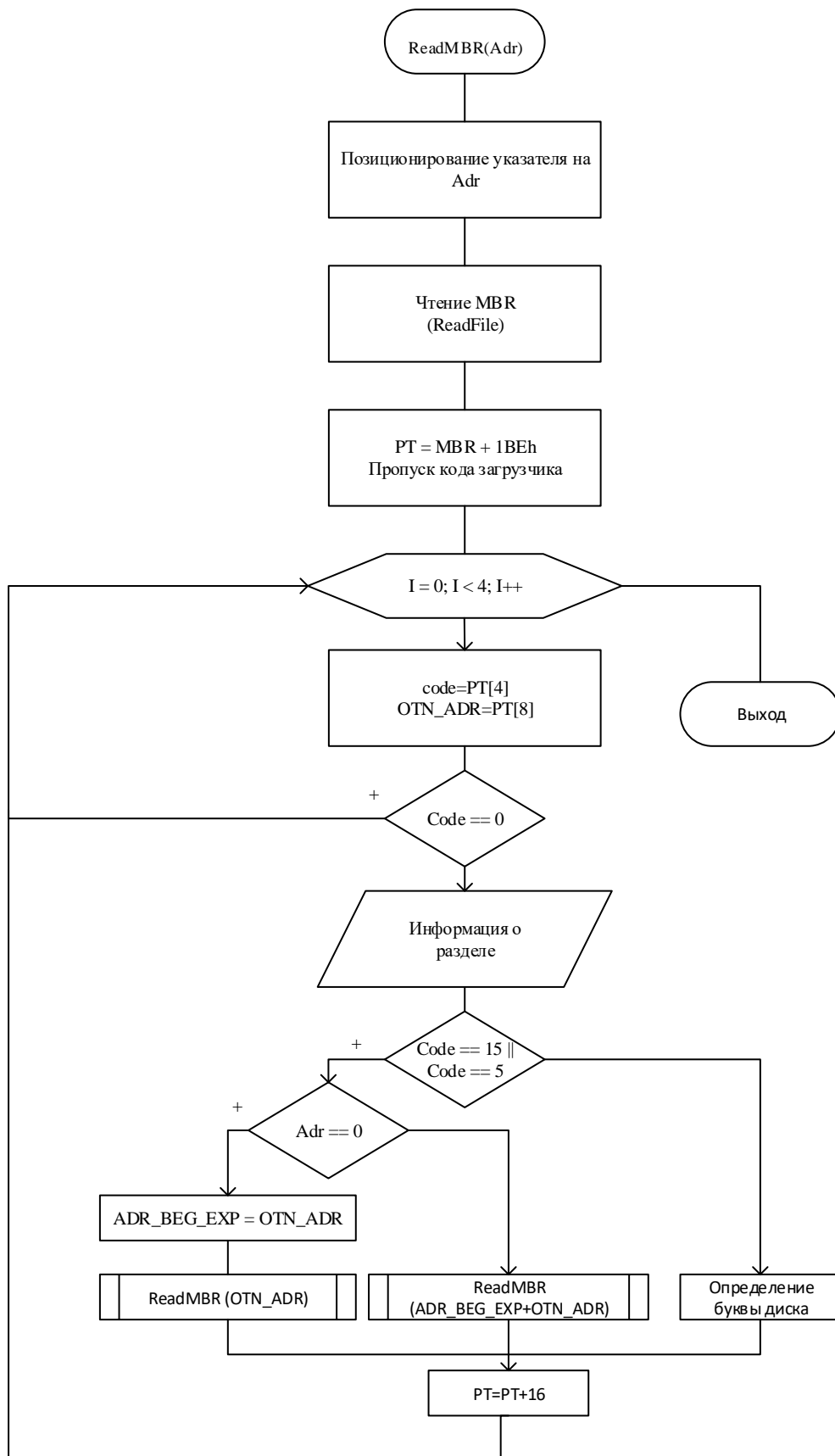


Рисунок 3 - Схема обработки данных MBR о разделах диска

Для реализации алгоритма использована среда разработки Microsoft Visual Studio 2019, исходный код программы написан на языке C# с использованием Windows Forms. Результат работы программы предоставлен на рисунке 4.

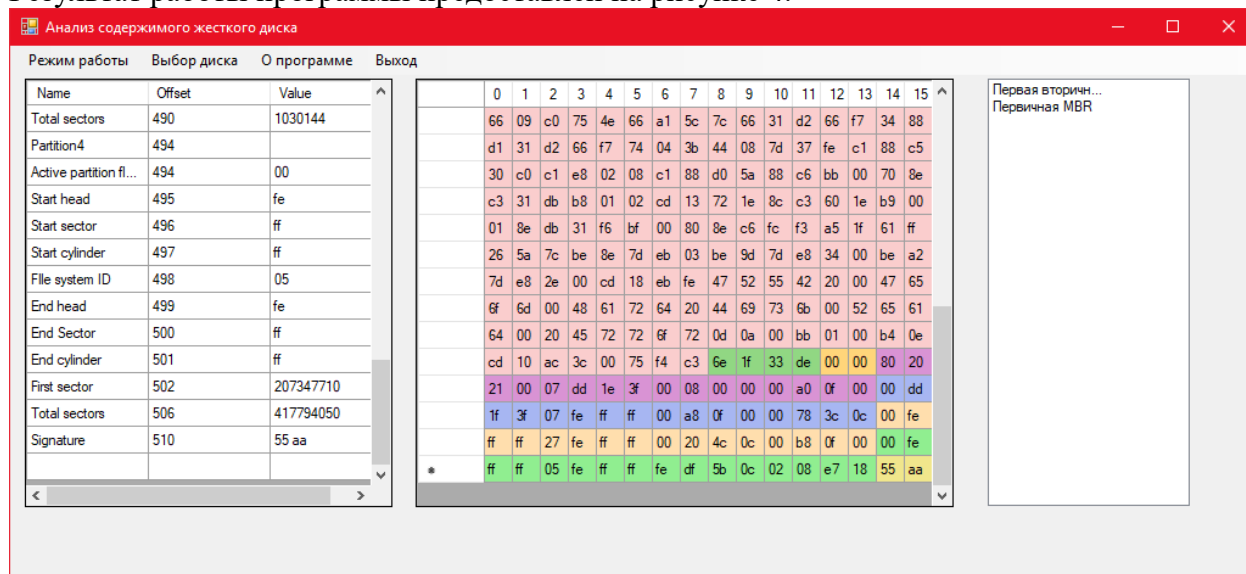


Рисунок 4 - Интерфейс утилиты анализа схемы разметки диска

В разработанном приложении показано содержимое MBR или GPT в шестнадцатеричных кодах, так как это сделано в популярной утилите DiskEditor. Различными цветами обозначены различные поля структур. Слева показаны поля структур, при наведении на определенное поле, выделяется и соответствующий этому полю шестнадцатеричный код. Особенностью разработанного приложения является возможность удобного перехода между первичной и вторичными MBR с помощью панели справа (рис. 4).

Выводы

Разработана программа чтения информации из главной загрузочной записи жесткого диска и определения всех логических дисков системы. Внедрение данной подсистемы в образовательный процесс позволит более детально изучить структуру таблицы разделов MBR и GPT, что позволит повысить знания студентов в области системного программного обеспечения.

Литература

1. Кэрриэ Б. Криминалистический анализ файловых систем [Текст]. – СПб.: Питер, 2007. – 480 с.: ил.
2. Фёдоров А. Жесткий диск изнутри: загрузка и таблицы разделов [Электронный ресурс]. – URL: <https://compress.ru/article.aspx?id=11179> (дата обращения: 20.04.2019).
3. Черногриз В.А. Главная Загрузочная Запись - Master Boot Record (MBR) [Электронный ресурс]. – URL: http://masters.donntu.org/2008/fvti/chernoriz/library/mbr_art.htm (дата обращения: 20.04.2019).
4. Смит Р. Раскройте весь потенциал дисков большой емкости с помощью GPT и Linux [Электронный ресурс]. – URL: <https://www.ibm.com/developerworks/ru/library/l-gpt/index.html> (дата обращения: 20.04.2019).

УДК 004.45

СТРУКТУРА ХРАНЕНИЯ ИНФОРМАЦИИ О СОСТАВЕ КАТАЛОГА В ФАЙЛОВОЙ СИСТЕМЕ NTFS

Чередникова О. Ю., Уткин П. С.

Донецкий национальный технический университет

кафедра компьютерной инженерии

E-mail: hawk1ua@gmail.com

Аннотация:

Чередникова О.Ю., Уткин П.С. Структура хранения информации о составе каталога в файловой системе NTFS. Выполнен обзор утилит, позволяющих просмотреть структуру информации в файловой системе NTFS. Проанализирована структура индексных записей. Разработано приложение, позволяющее просматривать информацию в NTFS-разделе.

Annotation:

Cherednikova O.Yu., Utkin P.S. The structure of storing information about the composition of the directory in the NTFS file system. A review of utilities that allow you to view the structure of information in the NTFS file system has been completed. The structure of index records is analyzed. An application that allows you to view information in the NTFS-section is developed.

Общая постановка проблемы

Для специалистов, занятых разработкой утилит для операционных систем, и, в частности файловых систем, важно понимать структуру хранения информации на логическом диске. Использование знаний о структуре существующих файловых систем также позволяет создавать новые файловые системы, используя преимущества имеющихся. Одной из самых используемых файловых систем операционных систем семейства Windows NT на сегодняшний день, является файловая система NTFS. Однако ее структуру нельзя назвать простой. В частности, способ хранения информации о составе директории в виде В-дерева, состоящего из индексных записей, не является очевидным. Поэтому актуальной является задача разработки программного приложения, позволяющего в удобном для пользователя виде отобразить структуру каталога NTFS-раздела.

Обзор утилит для анализа внутренней структуры NTFS-раздела

Из существующих решений одним из самых популярных является приложение DiskEditor (рис.1), которое обеспечивает чтение, запись, редактирование содержимого как физического, так и логического раздела. Информация выводится шестнадцатеричными кодами, а также разбита на полям в соответствии с выбранным шаблоном. Системные программисты могут использовать данную программу для понимания структуры данных и проверки правильности их реализации. Однако DiskEditor не может предоставить программисту полной структуры индексных записей каталога NTFS-раздела, т.е. не содержит шаблона для отображения информации в виде отдельных полей. Анализ структуры индексных записей с помощью этой утилиты приходится выполнять по шестнадцатеричным кодам.

Name	Offset	Value
Signature (must be 'FILE')	000	FILE
Offset to the update sequence	004	0x30
Update sequence size in words	006	3
\$LogFile Sequence Number (L...	008	34.025.326.350
Sequence number	016	1
Hard link count	018	1
Offset to the first attribute	020	0x38
> Flags	022	01 00
Real size of the FILE record	024	472
Allocated size of the FILE record	028	1.024
Base FILE record	032	0
Next attribute ID	040	7
ID of this record	044	0
Update sequence number	048	82 04
Update sequence array	050	00 00 00 00
> Attribute \$10	056	
> Attribute \$30	152	
> Attribute \$80	256	
> Attribute \$B0	392	
End marker	464	0xFFFFFFFF

Рисунок 1 - Анализ структуры NTFS-раздела в DiskEditor

Так же существует программа DiskExplorer for NTFS, благодаря которой системный программист может увидеть содержимое индексных узлов каталога и получить ссылку на их записи в MFT-таблице (рис.2).

Name	Mft#	Size	Date	Attr	Subs
AEF_WB3 1A Listeni...	x151[x1]	3068377	14.11.2013 08:20:22	a_____	Save View
AEF_WB3 1A Pronu...	x152[x1]	1126825	14.11.2013 08:20:36	a_____	Save View
AEF_WB3 1A Pronu...	x153[x1]	623089	14.11.2013 08:20:43	a_____	Save View
AEF_WB3 1B Listeni...	x154[x1]	2152645	14.11.2013 08:21:08	a_____	Save View
AEF_WB3 1B Pronu...	x155[x1]	699400	14.11.2013 08:21:16	a_____	Save View
AEF_WB3 2A Listeni...	x156[x1]	2637616	14.11.2013 08:21:47	a_____	Save View
AEF_WB3 2A Pronu...	x157[x1]	1085959	14.11.2013 08:22:00	a_____	Save View
AEF_WB3 2B Listeni...	x158[x1]	4048744	19.11.2013 06:35:49	a_____	Save View
AEF_WB3 2B Pronu...	x159[x1]	558871	14.11.2013 08:22:55	a_____	Save View
AEF_WB3 3A Listeni...	x15A[x1]	3318994	14.11.2013 08:23:33	a_____	Save View
AEF_WB3 3A Pronu...	x15B[x1]	737764	14.11.2013 08:23:43	a_____	Save View
AEF_WB3 3A Pronu...	x15C[x1]	682303	14.11.2013 08:23:51	a_____	Save View
AEF_WB3 3B Listeni...	x15D[x1]	3562939	14.11.2013 08:24:31	a_____	Save View
AEF_WB3 3B Prepo...	x15E[x1]	534685	14.11.2013 08:24:39	a_____	Save View
AEF_WB3 3B Pronu...	x15F[x1]	549697	14.11.2013 08:24:46	a_____	Save View
AEF_WB3 3B Pronu...	x160[x1]	672295	14.11.2013 08:24:54	a_____	Save View
AEF_WB3 4A Listeni...	x161[x1]	3593797	14.11.2013 08:25:35	a_____	Save View
AEF_WB3 4A Pronu...	x162[x1]	584308	14.11.2013 08:25:42	a_____	Save View
AEF_WB3 4B Listeni...	x163[x1]	3680533	14.11.2013 08:26:23	a_____	Save View
AEF_WB3 4B Pronu...	x164[x1]	440443	14.11.2013 08:26:29	a_____	Save View
AEF_WB3 4B Pronu...	x165[x1]	530098	14.11.2013 08:26:35	a_____	Save View
AEF_WB3 5A Listeni...	x166[x1]	3570028	14.11.2013 08:27:18	a_____	Save View
AEF_WB3 5A Pronu...	x167[x1]	873706	14.11.2013 08:27:32	a_____	Save View
AEF_WB3 5B Listeni...	x168[x1]	3487045	14.11.2013 08:28:18	a_____	Save View
AEF_WB3 5B Pronu...	x169[x1]	714829	14.11.2013 08:28:26	a_____	Save View
AEF_WB3 5B Pronu...	x16A[x1]	768205	14.11.2013 08:28:37	a_____	Save View

Рисунок 2 - Представление индексных узлов каталога в программе DiskExplorer

Однако приложение DiskExplorer также не отображает структуру B-дерева. Поэтому целью работы является разработка программного обеспечения для просмотра и анализа NTFS раздела, отображения структуры индексных записей в виде B-деревьев.

Структура хранения информации о составе каталога в NTFS

Все данные в NTFS – это файлы и информация о них хранятся в MFT-таблице (аббревиатура с англ. Master File Table). Каждый файл или каталог в файловой системе представлен как запись в MFT-таблице. Размер записи составляет 1 Кбайт. Первые 42 байта содержат в себе заголовок записи MFT. Остальные байты записи хранят в себе атрибуты файла. Атрибут записи – это небольшая структура данных, которая служит для хранения к примеру имени, или содержимого файла (рис.3) [1].

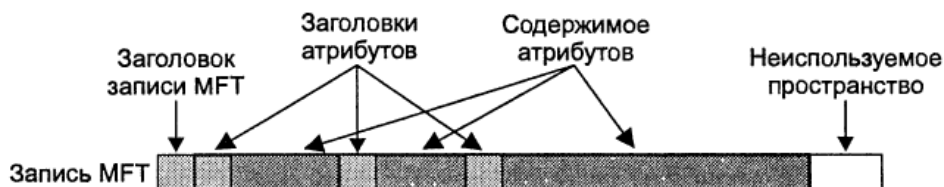


Рисунок 3 - Структура записи MFT в NTFS

Один из флагов в заголовке записи MFT показывает, что запись описывает каталог. Содержимое каталога сгруппировано в виде В-дерева.

В-дерево - сильноветвящееся сбалансированное дерево поиска, позволяющее проводить поиск, добавление и удаление элементов. В-дерево может применяться для структурирования информации (как правило, метаданных). Время доступа к произвольному блоку очень велико. Поэтому важно уменьшить количество узлов, просматриваемых при каждой операции. Использование поиска по списку каждый раз для нахождения случайного блока могло бы привести к чрезмерному количеству обращений к диску вследствие необходимости последовательного прохода по всем его элементам, предшествующим заданному, тогда как поиск в В-дереве, благодаря свойствам сбалансированности и высокой ветвистости, позволяет значительно сократить количество таких операций [2]. Принцип хранения информации о составе каталога в виде В-дерева следующий: узел дерева хранит имена файлов (атрибут \$FILE_NAME) отсортированными. Узел может иметь дочерние узлы, количество которых всегда на единицу больше, чем количество файлов в узле (рис.4) [3]. Самый левый дочерний узел содержит в себе имена, которые в отсортированном списке предшествуют первому имени файла узла. Следующий дочерний узел содержит имена файлов, находящихся в диапазоне между первым и вторым элементом родительского узла в отсортированном списке имен файлов каталога и т.д.

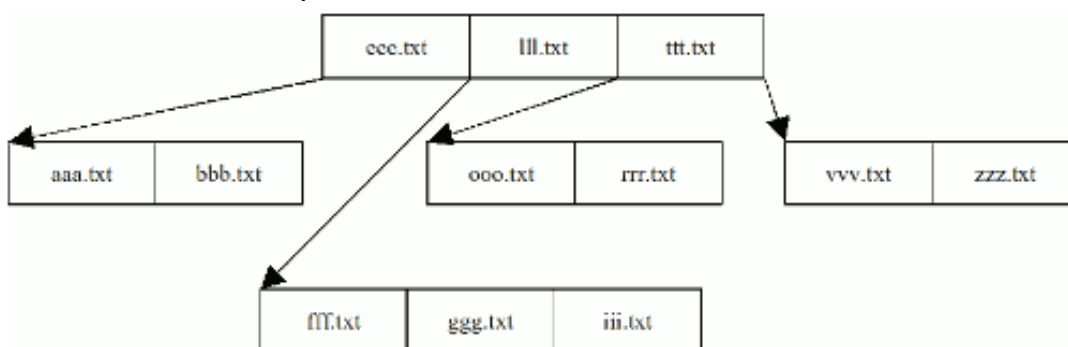


Рисунок 4 - В-дерево каталога NTFS

Для реализации В-дерева в NTFS используется понятие индекса, которое пришло из баз данных. Индекс - это коллекция элементов, хранящихся в отсортированном порядке. В качестве индексного элемента может выступать любой атрибут, по которому будет производиться индексация. Для индексов каталогов это всегда атрибут \$FILE_NAME, для каждого файла или каталога, содержащегося в нем. На каждый файл или каталог приходится

как минимум две структуры FILE_NAME, первый как атрибут у файла, а второй, используемый для индекса. Узел дерева хранит последовательность атрибутов \$FILE_NAME [4].

Для хранения узлов дерева используются два типа атрибутов \$INDEX_ROOT, который присутствует всегда для любой директории и \$INDEX_ALLOCATION, который может и не присутствовать для небольших каталогов.

Атрибут \$INDEX_ROOT представляет собой корневой узел индексного дерева. Индексное дерево состоит из индексных элементов, которые являются файлами или каталогами. Атрибут INDEX_ALLOCATION не ограничен по количеству узлов и поэтому он будет содержать индексные элементы, которые не поместились в INDEX_ROOT.

Тело атрибута \$INDEX_ALLOCATION хранит индексные элементы в индексных записях, каждая из которых обладает заголовком. Индексная запись имеет статический размер и представляет один узел дерева. Ее размер указывается либо в поле в \$INDEX_ROOT либо в поле загрузочного сектора, где может быть байтовый размер записи или двоичный логарифм размера.

Разработка приложения анализа структуры директории

Разработано приложение для анализа информации каталогов NTFS-раздела на языке программирования С#. Для чтения информации с логического диска использовались функции Windows API (CreateFile, ReadFile, SetFilePointer). Алгоритм приложения включает в себя следующие этапы:

1) Чтение первого сектора логического диска для определения расположения MFT-таблицы и других параметров раздела.

2) Организация цикла чтения записей MFT. Для каждой записи производится анализ ее заголовка и атрибутов.

3) Определение записи, принадлежащей каталогу и считывание его атрибута \$INDEX_ROOT. Перебор всех элементов индексного узла. Для перехода от одного элемента к другому следует к адресу индексного элемента добавлять смещение следующего элемента относительно текущего.

4) Определение наличия дочерних узлов В-дерева и считывание атрибута \$INDEX_ALLOCATION. Атрибут представляет собой последовательность индексных записей, количество которых может быть вычислено делением размера атрибута на размер индексной записи. Зная количество индексных записей, их можно обойти как линейный массив элементов. Каждая индексная запись состоит из индексных узлов, которые просматриваются аналогично индексным узлам атрибута \$INDEX_ROOT.

В ходе разработки были использованы основные особенности ООП, такие как классы, методы, интерфейсы, инкапсуляция, полиморфизм и наследование. Функциональные возможности разработанного приложения включают в себя чтение MFT-таблицы, перемещаться по записям MFT-таблицы, а так демонстрацию пользователю структуры индексных записей каталога в виде списка.

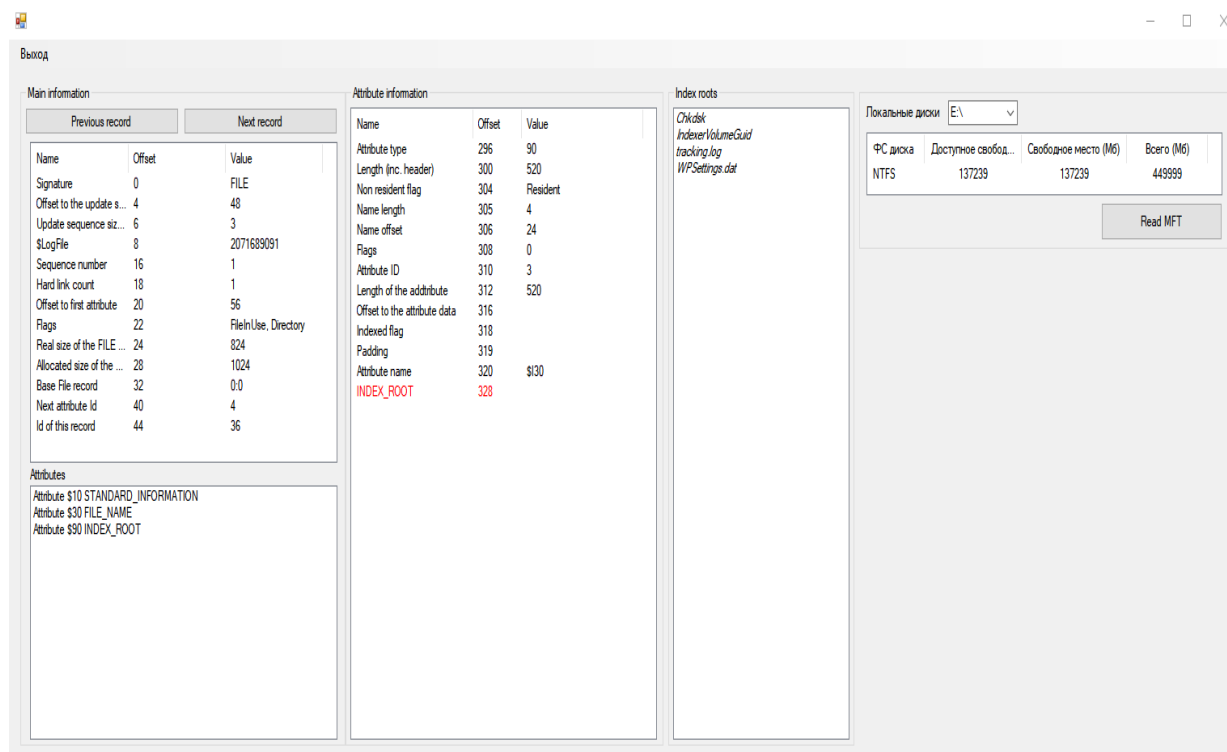


Рисунок 5 - Интерфейс программы для чтения NTFS раздела и индексных узлов каталогов

Заключение

Разработанное приложение позволяет в удобном для пользователя виде увидеть структуру информации на логическом диске с файловой системой NTFS. Приложение может быть использовано студентами IT-специальностей при изучении дисциплины «Операционные системы», а также специалистами в области системного программного обеспечения для анализа структуры информации в файловой системе NTFS. Дальнейшие исследования и разработки связаны со структуризацией списка индексных записей каталога в виде В-дерева, переходом к связанной MFT-записи из узла В-дерева, улучшением интерфейса пользователя.

Литература

1. Кэрриэ Б. Криминалистический анализ файловых систем [Текст]. – СПб.: Питер, 2007. – 480 с.: ил.
2. Руссинович М., Соломон Д. Внутреннее устройство Microsoft Windows. 6-е изд. — СПб.: Питер, 2013. — 800 с.: ил. — (Серия «Мастер-класс»)
3. Кастер Х. Основы Windows NT и NTFS : [Пер. с англ.] / Хелен Кастер. - М. : Изд. отд. "Рус. редакция", 1996. - 436 с. : ил.
4. Таненбаум Э., Бох С. Современные операционные системы. 4-е изд. - СПб.: 2015. — 1120с.

УДК 004.42, 004.62, 004.51

ВИДЫ НЕЙРОННЫХ СЕТЕЙ И СПОСОБЫ ИХ ОБУЧЕНИЯ

Щёголев В. А., Чередникова О. Ю.

Донецкий Национальный Технический университет, г. Донецк, Донецкая Народная Республика

Аннотация

Щёголев В. А., Чередникова О. Ю. Виды нейронных сетей и способы их обучения.

В статье дано толкование понятий нейронные сети и искусственные нейронные сети, выполнен анализ существующих методов обучения нейронных сетей. Приведена классификация видов искусственных нейронных сетей.

Ключевые слова: *Нейронные сети, искусственные нейронные сети, методы обучения.*

Annotation

Shogolev V. A., Cherednikova O. Y. The types of neural networks and methods of their training. *The article gives an interpretation of the concepts of neural networks and artificial neural networks, analyzes the existing methods of teaching neural networks. The classification of types of artificial neural networks is given.*

Key words: *Neural networks, artificial neural networks, training methods.*

Введение

Нейронные сети – это одно из направлений исследований в области искусственного интеллекта, направленное на воспроизведение нервной системы человека. А именно: способность нервной системы обучаться и исправлять ошибки.

Нейронные сети (искусственная нейронная сеть) — это система соединенных и взаимодействующих между собой простых процессоров. Такие процессоры обычно довольно просты. Каждый процессор подобной сети имеет дело только с сигналами, которые он периодически получает, и сигналами, которые он периодически посылает другим процессорам. И, тем не менее, будучи соединёнными в достаточно большую сеть с управляемым взаимодействием, эти процессоры вместе способны выполнять довольно сложные задачи.

Способы обучения нейронных сетей

1. **Обучение с учителем (supervised learning)** предполагает наличие полного набора размеченных данных для тренировки модели на всех этапах ее построения.

Наличие полностью размеченного датасета означает, что каждому примеру в обучающем наборе соответствует ответ, который алгоритм и должен получить. Таким образом, размеченный датасет из фотографий цветов обучит нейронную сеть, где изображены розы, ромашки или нарциссы. Когда сеть получит новое фото, она сравнит его с примерами из обучающего датасета, чтобы предсказать ответ. [1]

В основном обучение с учителем применяется для решения двух типов задач: классификации и регрессии.

В задачах классификации алгоритм предсказывает дискретные значения, соответствующие номерам классов, к которым принадлежат объекты. В обучающем датасете с фотографиями цветов каждое изображение будет иметь соответствующую метку — «ромашка», «роза» или «лютик». Качество алгоритма оценивается тем, насколько точно он может правильно классифицировать новые фото с ромашками и черпаками.

Задачи регрессии связаны с непрерывными данными. Один из примеров, линейная регрессия, вычисляет ожидаемое значение переменной y , учитывая конкретные значения x .

2. Обучение без учителя

Идеально размеченные и чистые данные достать нелегко. Поэтому иногда перед алгоритмом стоит задача найти заранее не известные ответы. Вот где нужно обучение без учителя.

В обучении без учителя (**unsupervised learning**) у модели есть набор данных, и нет явных указаний, что с ним делать. Нейронная сеть пытается самостоятельно найти корреляции в данных, извлекая полезные признаки и анализируя их.

В зависимости от задачи модель систематизирует данные по-разному.

- Кластеризация. Наиболее распространенная задача для обучения без учителя. Алгоритм подбирает похожие данные, находя общие признаки, и группируют их вместе.
- Ассоциации. Некоторые характеристики объекта коррелируют с другими признаками. Рассматривая пару ключевых признаков объекта, модель может предсказать другие, с которыми существует связь.
- Автоэнкодеры принимают входные данные, кодируют их, а затем пытаются воссоздать начальные данные из полученного кода. Не так много реальных ситуаций, когда используют простой автоэнкодер. Но стоит добавить слои и возможности расширятся: используя зашумленные и исходные версии изображений для обучения, автоэнкодеры могут удалять шум из видеоданных, изображений или медицинских сканов, чтобы повысить качество данных.

В обучении без учителя сложно вычислить точность алгоритма, так как в данных отсутствуют «правильные ответы» или метки. Но размеченные данные часто ненадежные или их слишком дорого получить. В таких случаях, предоставляя модели свободу действий для поиска зависимостей, можно получить хорошие результаты.

3. Обучение с частичным привлечением учителя. [2]

Обучение с частичным привлечением учителя (**semi-supervised learning**) характеризуется своим названием: обучающий датасет содержит как размеченные, так и неразмеченные данные. Этот метод особенно полезен, когда трудно извлечь из данных важные признаки или разметить все объекты – трудоемкая задача.

Этот метод машинного обучения распространен для анализа медицинских изображений, таких как сканы компьютерной томографии или МРТ. Опытный рентгенолог может разметить небольшое подмножество сканов, на которых выявлены опухоли и заболевания. Но вручную размечать все сканы — слишком трудоемкая и дорогостоящая задача. Тем не менее нейронная сеть может извлечь информацию из небольшой доли размеченных данных и улучшить точность предсказаний по сравнению с моделью, обучающейся исключительно на неразмеченных данных.

Популярный метод обучения, для которого требуется небольшой набор размеченных данных, заключается в использовании генеративно-сопоставительной сети или GAN.

Представьте себе соревнование двух нейронных сетей, где каждая пытается перехитрить другую. Это GAN. Одна из сетей, генератор, пытается создать новые объекты данных, которые имитируют обучающую выборку. Другая сеть, дискриминатор, оценивает, являются ли эти сгенерированные данные реальными или поддельными. Сети взаимодействуют и циклично совершенствуются, поскольку дискриминатор старается лучше отделять подделки от оригиналов, а генератор пытается создавать убедительные подделки.

4. Обучение с подкреплением

Обучение с подкреплением (**reinforcement learning**) действует по принципу игры, при достижении конкретной цели получает награду. Видеоигры — популярная тестовая среда для исследований.

Агенты ИИ пытаются найти оптимальный способ достижения цели или улучшения производительности для конкретной среды. Когда агент предпринимает действия, способствующие достижению цели, он получает награду. Глобальная цель — предсказывать следующие шаги, чтобы заработать максимальную награду в конечном итоге.

При принятии решения агент изучает обратную связь, новые тактики и решения способные привести к большему выигрышу. Этот подход использует **долгосрочную стратегию** — так же как в шахматах: следующий наилучший ход может не помочь выиграть в конечном счете. Поэтому агент пытается максимизировать суммарную награду.

Чем больше уровней с обратной связи, тем лучше становится стратегия агента. Такой подход особенно полезен для обучения роботов, которые управляют автономными транспортными средствами или инвентарем на складе.

Виды искусственных нейронных сетей

Нейросеть состоит из двух основных слоёв - принимающего (он же и распределительный) сигналы и обрабатывающего. Однако, если нейронная сеть состоит только лишь из этих двух слоев - то она однослойная, если слоев больше, то многослойная [3]. Однослойная ИНС (рис.1) - входящие сигналы сразу передаются с входного слоя на выходной, который обрабатывает их и выдает готовый результат. На изображении распределяющий слой изображен кружочками, а обрабатывающий - квадратиками.

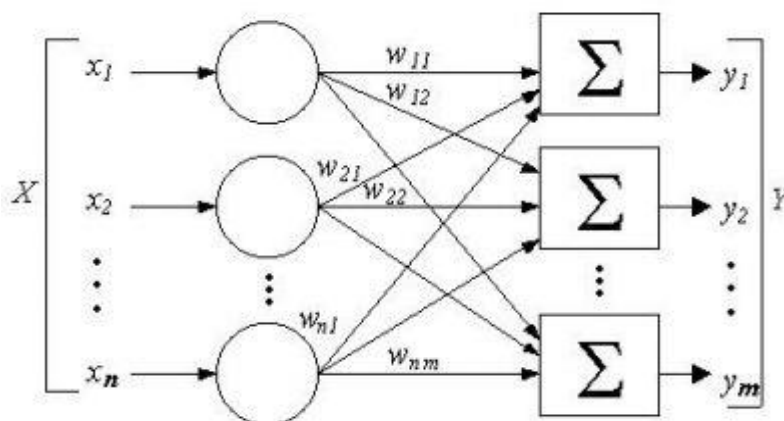


Рисунок 1 – Однослойная нейронная сеть

Многослойная ИНС (рис.2) - сеть, которая состоит из входного, скрытого и обрабатывающего слоев. Сигнал с распределяющего слоя частично обрабатывается скрытым слоем, после чего передается на последний слой нейронов, вычисляющий конечный результат.

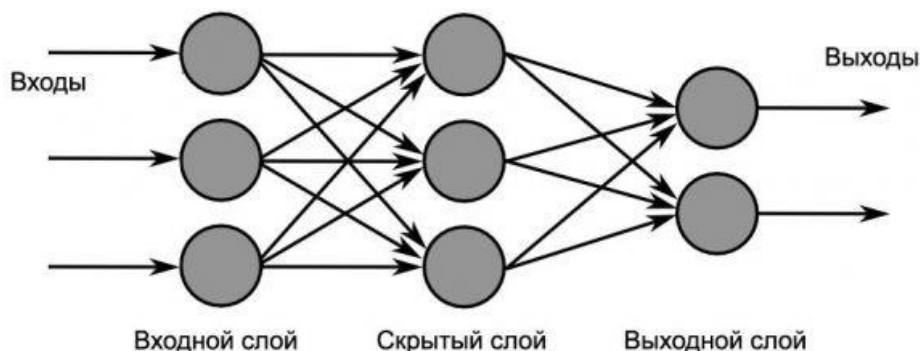


Рисунок 2 – Многослойная нейронная сеть

Скрытые слои ИНС ученые научились обучать совсем недавно и это большой шаг вперед, поскольку многослойные нейросети существенно превосходят по производительности и возможностям однослойные.

Сети работают в двух направлениях - прямое распределение и обратное. ИНС прямого распределения дают возможность решать с успехом большинство задач: прогнозирование, кластеризация и распознавание. В таких нейросетях сигнал передается только вперед, назад он возвращаться не имеет возможности.

В подобных сетях часть сигнала от нейронов может быть возвращена назад и такой принцип работы существенно расширяет возможности нейросетей. Такие ИНС могут обладать кратковременной памятью как у человека. (рис.3)

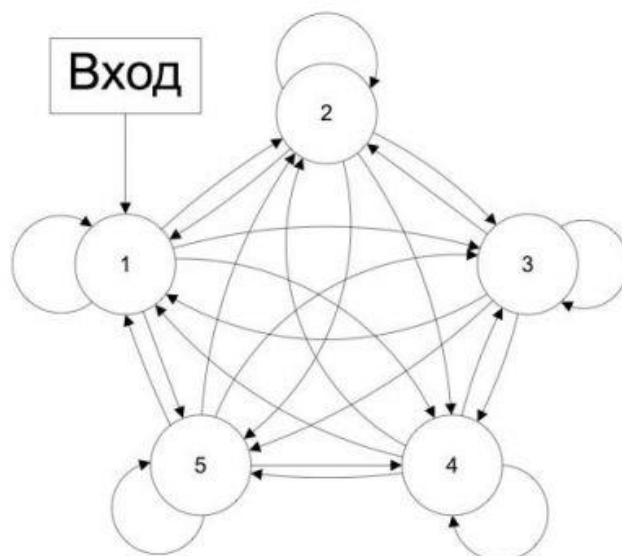


Рисунок 3 – Обратные связи ИНС

Выводы

В статье рассмотрены способы обучения нейронных сетей. Каждый вид обучения имеет свои уникальные особенности, которые нужно учитывать при выборе метода обучения нейронной сети для решения поставленной задачи.

Был выполнен анализ существующих видов искусственных нейронных сетей. А так же приведена структура их реализации.

Литература

1. Ежов А. А, Шумский С. А. *Нейрокомпьютинг и его применение в экономике и бизнесе*. М.: МИФИ, 1998. 224 с.
2. Laurene Fausett, Laurene V. Fausett. *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications* Prentice-Hall, 1994. 461 с.
3. Ф. Уоссермен, *Нейрокомпьютерная техника: Теория и практика*. 1992 г.

УДК 004.3

ПРИМЕНЕНИЕ ЧПУ ДЛЯ ИЗГОТОВЛЕНИЯ ЗАКАЗНЫХ ПЕЧАТНЫХ ПЛАТ

Захарченко К. С., Достлев Ю. С.

Донецкий национальный технический университет,
кафедра компьютерной инженерии

E-mail: zakharchenko.kostya@gmail.com

Аннотация:

Захарченко К. С., Достлев Ю. С. Применение ЧПУ для изготовления заказных печатных плат. В статье проведен анализ основных способов производства печатных плат (ПП) в ориентации на рациональность их применения в условиях одиночного или мелкосерийного производства. Обоснован подход к производству малогабаритных плат как рациональных для участков – лабораторий на этапе проектирования и исследования при проектировании новых радиотехнических или вычислительных устройств. Предложен вариант конструктивного решения для устройства нанесения рисунка на базе специализированного двухкоординатного графопостроителя.

Ключевые слова: шаговый привод, число-программное управление, контроллер, печатная плата.

Annotation:

Zakharchenko K.S., Dostlev Yu.S. The use of CNC for the manufacture of custom printed circuit boards. The article analyzes the main methods of production of printed circuit boards (PCB) in the orientation on the rationality of their use in single or small-scale production. The approach to the production of small boards as rational for sites - laboratories at the design stage and research in the design of new radio or computing devices has been substantiated. A variant of a constructive solution for the device for drawing a picture on the basis of a specialized two-coordinate plotter has been proposed.

Keywords: stepper drive, number-program control, controller, printed circuit board.

Введение

В настоящее время в большинстве случаев для монтажа радиотехнических устройств и блоков вычислительной техники конструктивно используется печатный монтаж с минимальным числом длины проводников. Для использования поверхностного монтажа на базе СИМД элементов, к печатным платам предъявляются повышенные требования по точности, но при этом в ряде случаев удается сократить число слоев ПП до односторонней или двухсторонней. При проектировании таких печатных плат большое значение имеет возможность оперативного изготовления опытных экземпляров плат для проведения исследований и настройки схем с обеспечением возможности их получения в лабораторных условиях с минимальными затратами.

Таким образом, актуальной задачей является создание устройств, позволяющих решать эти вопросы. Данная работа посвящена разработке устройства нанесения рисунка на заготовки ПП в лабораторных условиях.

Анализ методов производства заготовок печатных плат

Применяемые в настоящее время методы изготовления печатных плат (ПП) можно объединить в три группы [1].

Субтрактивные, или химические, методы предполагают получение рисунка проводников за счет удаления участков фольги с пробельных мест в основном химическим способом – травлением. Внутри группы методы подразделяются по способу нанесения защитного рисунка: на фотохимический, сеточно-химический, офсетно-химический. К этой

группе относится и метод механического гравирования - удаление фольги между проводниками торцевой фрезой малого диаметра на станках с числовым программным управлением (ЧПУ). Недостатком этих методов является отсутствие металлизированных отверстий. Применяются они для производства не сложных ОПП.

Аддитивные методы предусматривают нанесение рисунка проводников на диэлектрическое (нефольгированное) основание путем электрохимического осаждения меди и металлизацию отверстий в одном технологическом процессе. К этой группе относятся: собственно аддитивный (или аддитивный химический) метод, в котором проводники и металлизированные отверстия получаются только за счет химического осаждения меди из раствора солей меди (без гальванического наращивания); полуаддитивный (или электрохимический) метод, в котором химическое осаждение тонкого слоя меди проводится на всю поверхность платы, затем на проводниках и в отверстиях слой меди наращивается гальванически, а в конце процесса химически осажденная медь удаляется с пробельных мест травлением.

Отличительной особенностью этих методов является особый вид подготовки поверхности для улучшения адгезии осажденной меди, заключающийся в нанесении на поверхность диэлектрика специальных составов – адгезивов. Адгезив может наноситься на заводе, изготавливающем материал основания ПП. Аддитивные методы дают большую экономию меди и высокую разрешающую способность [1].

Комбинированные методы получаются от сочетания химического и электрохимического методов. Для ПП берут фольгированное с двух сторон основание, рисунок проводников получают травлением фольги, а монтажные и переходные отверстия металлизуются электрохимическим методом.

Комбинированные методы бывают двух видов: негативный и позитивный, которые отличаются друг от друга порядком выполнения операций. Наибольшее распространение получил комбинированный позитивный метод с применением сухих пленочных фоторезисторов.

Каждый из приведенных методов имеет свои достоинства и недостатки и может быть эффективно применен в конкретных условиях производства.

Выбор и обоснование класса точности печатной платы

По точности изготовления элементов печатного монтажа ПП делят на 5 классов. Наименьшие номинальные значения основных размеров элементов конструкции ПП в зависимости от классов точности приведены ниже в ГОСТ 23751-86 (таблица 1).

Таблица 1— Основные параметры и размеры ПП [2]

Класс точности	Минимальная ширина проводника, мм	Расстояние между краями соседних элементов, мм	Гарантийный пояс, мм	γ^*
1	0.75	0.75	0.3	0.4
2	0.45	0.45	0.2	0.4
3	0.25	0.25	0.1	0.33
4	0.15	0.15	0.05	0.25
5	0.1	0.1	0.025	0.2

* γ — отношение номинального значения диаметра наименьшего из металлизированных отверстий к толщине ПП.

Платы 1-го и 2-го классов характеризуются простотой исполнения, низкой стоимостью, высокой надежностью, но имеют большие габариты. ПП 3-го класса —

наиболее распространенные, поскольку обеспечивают достаточно высокую плотность трассировки и монтажа, а также для их производства требуется рядовое оборудование. ПП 4-го класса выпускаются на высокоточном оборудовании, но требования к материалам, оборудованию и помещениям ниже, чем для пятого класса.

Изготовление ПП 5-го класса требует применения уникального высокоточного оборудования, дорогих материалов, безусадочной фотопленки и даже создания в производственных помещениях «чистой зоны» с термостатированием. Но ПП небольшого размера могут выполняться по пятому классу на оборудовании, обеспечивающем получение плат четвертого класса [2].

Анализ способов подготовки к субтрактивному производству ПП

Стандартный графопостроитель имеет большие габариты рабочей области, но для опытного производства это не требуется. Основной недостаток – сложность позиционировать фольгированную заготовку на плоскости графопостроителя без вмешательства в конструкцию графопостроителя, а это в большинстве случаев будет приводить к нарушению возможности в дальнейшем использовать его как стандартный плоттер.

Специализированное устройство рационально проектировать для оперативной подготовки ПП опытных радиотехнических устройств или вычислительных блоков специализированных систем контроля и управления, имеющих небольшие габариты. Это позволит при проектировании, отладке и настройке использовать модульный подход, то есть отдельные функциональные узлы конструктивно размещать на отдельных ПП. В процессе исследований, доработок и настройки такие выделенные конструктивно– функциональные модули легко заменять, не изменяя общей конструкции проектируемого устройства.

Разрабатываемое устройство должно иметь достаточно малые габариты, обладать высокой надежностью работы. Для реализации определенного класса точности изготовления ПП следует применять пишущее устройство с толщиной пера достаточной для нанесения проводников минимальной ширины. Изменяя толщину пера пишущего устройства можно обеспечить настройку точности изготовления рисунка проводников ПП и обеспечить гибкость в выборе класса.

Конструктивные особенности устройства получения рисунка

Основные требования к созданию устройства нанесения рисунков проводников на поверхность заготовок для химического или электрохимического производства ПП:

- максимальное использование готовых конструктивных узлов;
- возможность формирования общей конструкции в лабораторных условиях;
- параметры устройства по точности нанесения рисунка должны удовлетворять выбранному классу точности;
- возможность получения позитивных и негативных рисунков проводников проектируемых ПП;
- обеспечение возможности повторного нанесения рисунка с позиционированием над ранее полученным изображением ПП.

Все эти требования может обеспечить устройство, в виде двухкоординатного компьютерно-управляемого графопостроителя со специальной конструкцией крепления заготовок и использованием различных рисующих приспособлений.

Согласно требованию максимально использовать готовые конструктивные узлы, двигатели для устройства будут взяты от не актуальных на сегодня приводов CD-ROM (рис.1).



Рисунок 1 – Шаговый двигатель CD-ROM

В качестве контроллера для устройства выбрана плата Arduino UNO. Arduino UNO - самый распространенный вариант контроллеров от Arduino, он выполнен на микроконтроллере ATmega328p [3]. Его характеристик будет достаточно для проектируемой системы. Управление двигателями будет осуществляться через драйвер шаговых двигателей модели A4988. Подключить драйвер к плате Arduino Uno поможет плата-расширитель CNC Shield v3 (рис.2).



Рисунок 2 – Arduino CNC Shield v3

Данная плата предоставляет собой интерфейс для подключения драйвера A4988 к плате Arduino.

Выводы

Обоснован подход к производству малогабаритных плат как рациональных для участков – лабораторий на этапе проектирования и исследования при проектировании новых радиотехнических или вычислительных устройств. Предложен вариант конструктивного решения для устройства нанесения рисунка на базе специализированного двухкоординатного графопостроителя.

Литература

1. Методы изготовления печатных плат. [Электронный ресурс]. – URL: <https://pcbdesigner.ru/pcb/sposobi-izgotovleniya-pechatnih-plat/metody-izgotovleniya-pechatnyx-plat.html>
2. ГОСТ 23751-86. [Электронный ресурс]. – URL: <http://www.lamsystems-lto.ru/files/pdf/gost-23751.pdf>
3. Джерими Блум: Изучаем Arduino: инструменты и методы технического волшебства. Пер. с англ. - СПб.: БХВ-Петербург, 2015. – 336 с.

УДК 681.586.48

ОСОБЕННОСТИ ПРИМЕНЕНИЯ ДАТЧИКА ТЕМПЕРАТУРЫ И ВЛАЖНОСТИ DHT11

Евтушенко О. В., Кривошеев С. В.

Донецкий национальный технический университет,
кафедра компьютерной инженерии

E-mail: evtushenko.mai.ru@mail.ru

Аннотация

Евтушенко О. В., Кривошеев С. В. Особенности применения датчика температуры и влажности DHT11. В данной статье рассмотрены сферы применения датчиков температуры. Выполнен анализ характеристик и принципа функционирования датчика DHT11. Разработана схема подключения к Arduino.

Ключевые слова: Arduino, DHT11, датчик температуры и влажности.

Annotation

Evtushenko O. V., Kryvosheyev S. V. Features of the temperature and humidity sensor DHT11 application. This article discusses the scope of temperature sensors. The characteristics and operation principle of the DHT11 sensor are analyzed. The connection scheme to Arduino is developed.

Key words: Arduino, DHT11, temperature and humidity module.

Введение

Применение датчиков температуры и влажности охватывает большинство сфер производства, где температура и влажность оказывает влияние на качество работы, которая требует напряженного контроля. К основным таким сферам производства относятся следующие:

- топливная промышленность;
- металлургическая промышленность;
- машиностроение;
- транспортная отрасль;
- сельское хозяйство.

В каждой сфере используется определенная разновидность датчиков температуры и влажности, каждая из которых отличается своими особенностями и адаптацией к ним. К примеру, любое транспортное средство имеет в себе массу механических, электромеханических и электронных составляющих, оптимальная работа которых должна осуществляться независимо от внешних условий, для этого используются датчики, которые фиксируют температуру, давление воздуха и др. Также для большинства технологических процессов необходимо поддерживать определенный микроклимат, для этого используют датчики влажности. Помимо этого, эти датчики используют для регулировки сухости воздуха в помещении, а также для измерения насыщения почвы водой. Из-за того, что влагосодержание зависит от температуры, т.е. чем выше температура, тем выше содержание воды в том же объеме газа, в датчики влажности включают датчик температуры.

Выбор и анализ объекта исследования

По принципу работы датчики температуры и влажности можно разделить на следующие:

1. Емкостный – датчик с диэлектриком, свойства которого изменяются в зависимости от количества водяных паров в нём.
2. Резистивный – датчик, основой которого является изменения сопротивления, из-за количества влаги в нём.
3. Термисторный – датчик с парой однотипных термисторов, сопротивление которых прямо пропорционально температуре. Термисторы связаны по мостовой схеме, на одну диагональ подается питание, с другой получают значения влажности.
4. Оптический – самый точный датчик, в котором влажность фиксируется с помощью термина «точка росы».

В данной работе будет рассматриваться сенсор с емкостным датчиком и процесс сорбции в нем.

Общая характеристика датчика температуры и влажности DHT11

Сенсор DHT11 является цифровым датчиком, который содержит терморезистор и емкостный датчик влажности, предназначенные для измерения температуры и относительной влажности воздуха соответственно. Он включает в себя аналого-цифровой преобразователь для преобразования зафиксированных показателей влажности и температуры. Данный датчик не отличается высокой точностью, однако он несложен и отлично подходит для проверки температуры и влажности, к примеру в помещении. [1]

Схема подключения цифрового датчика температуры и влажности DHT11 к Arduino приведена на рисунке.

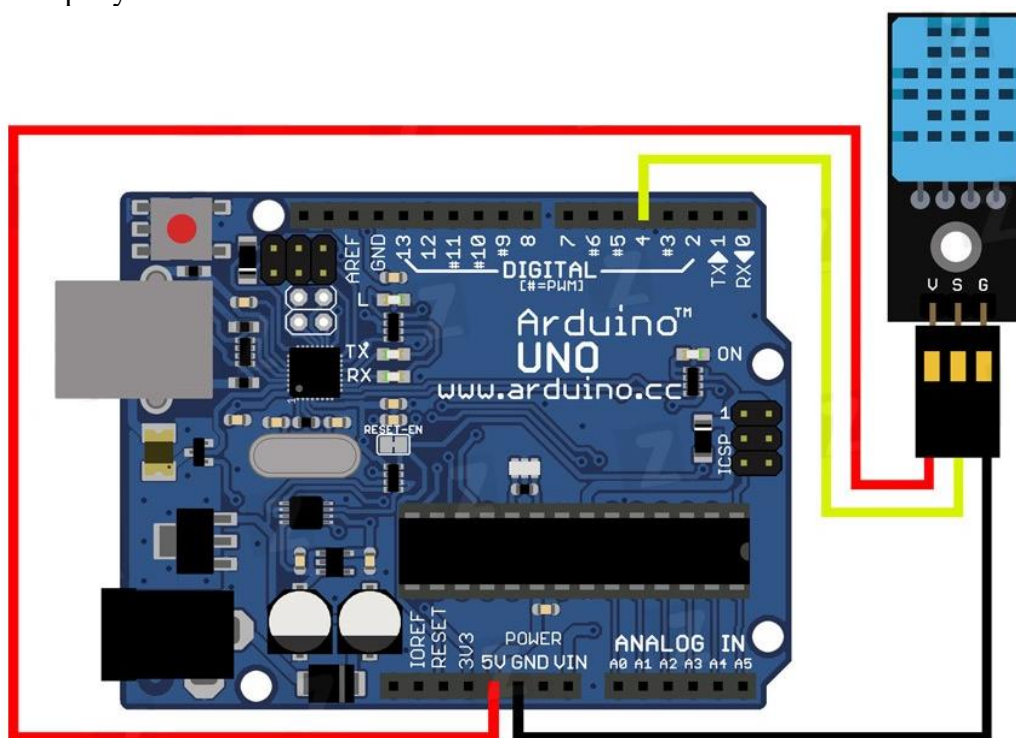


Рисунок – Схема подключения цифрового датчика температуры и влажности DHT11 к Arduino

Описание контактов: G – Подключается к выводу GND; V – Подключается к выводу +5V; S – Подключается к цифровому выводу.

Основные характеристики датчика DHT11:

1. В режиме измерения значения тока находится в диапазоне от 0.5 до 2.5 мА;
2. В режиме ожидания ток равен 0.15 мА;
3. Частота запроса не более одного раза в 1 с;
4. Максимальная длина экранированной линии связи 20 м;
5. Погрешность для фиксирования значений влажности при температуре до 25°C будет равна 4%, а при температуре от 25 до 50°C – 5%;
6. Диапазон относительной влажности при температуре:
 - 0°C равен от 30 до 90 %;
 - 25°C равен от 20 до 90 %;
 - 50°C равен от 20 до 80 %;
7. Максимальное время ответа датчика равно 15 с;
8. Гистерезис приблизительно равен 1 %.

Принцип действия датчика DHT11:

Сенсор DHT11 содержит цифровой сигнал. И температура, и влажность передаются по одному сигнальному проводу (S). Сенсор взаимодействует с Arduino по персональной шине. Связь двунаправленная и описать её можно следующим образом:

1. Микроконтроллер отправляет запрос о необходимости получить значения, это осуществляется с помощью установки сигнального провода сначала в ноль, а через некоторое время в единицу. В свою очередь сенсор должен подтвердить отправление значений, это осуществляется аналогично образом, сигнальный провод сначала устанавливается в ноль, а через некоторое время в единицу.

2. Далее сенсор отправляет вектор из нулей и единиц, последовательно создавая пять байтов, то есть сорок бит. В первых 16 битах будет находиться значение температуры, в вторых 16 битах – влажности и в последних 8 битах – контрольной суммы для проверки отсутствия ошибок при считывании.

Данный сенсор осуществляет измерения только по запросу и за счет этого достигается рациональное использование энергии: пока запроса на получение значений нет, сенсор потребляет очень малый ток [3].

Выводы

Был проанализирован датчик температуры и влажности DHT11. Была построена схема подключения к среде Arduino. А также построена математическая модель и исследована зависимость изменения емкости влагочувствительных элементов с разной основой.

Литература

1. Иго, Том Arduino, датчики и сети для связи устройств / Том Иго. - М.: БХВ-Петербург, 2011.
2. Подлепецкий Б. И., Симаков А. Б. Микроэлектронные датчики влажности, 1987.
3. Кашкаров, А. П. Датчики в электронных схемах. От простого к сложному / А.П. Кашкаров. - М.: ДМК Пресс, 2013.

УДК 681.586.48

ПРОГРАММНАЯ МОДЕЛЬ ТЕМПЕРАТУРНОГО ДАТЧИКА DS18B20

Симак А. С., Кривошеев С. В.

Донецкий национальный технический университет
кафедра компьютерной инженерии

Аннотация

Симак А. С., Кривошеев С. В. Программная модель температурного датчика DS18B20. В статье выполнен анализ характеристик, особенностей и принципа функционирования датчика DS18B20. Дана формула, описывающая зависимость погрешности от температуры. Построена программная модель с помощью ООП.

Ключевые слова: Arduino, DS18B20, датчик температуры, программно-имитационная модель DS18B20.

Annotation

Simak A. S., Kryvosheyev S. V. Software model of DS18B20 temperature sensor. The article analyzes the characteristics, features and principle of operation of the DS18B20 sensor. A formula describing the dependence of the error on temperature is given. The software model is built using OOP.

Key words: Arduino, DS18B20, temperature sensor, software simulation model DS18B20.

Введение

В современном мире мобильные компьютерные системы занимают важную роль в обеспечении работы интеллектуальных устройств.

Современные технологии позволяют разрабатывать устройства, обладающие следующими характеристиками:

- высокая производительность;
- низкое энергопотребление;
- мобильность.

Такие устройства не минули и автомобильную промышленность. Система для контроля дистанции между автомобилем и окружающими объектами приводит к снижению ДТП на дороге, автоматизированный кондиционер поддержит оптимальную температуру в салоне автомобиля, а устройство навигации поможет найти самый рациональный маршрут до той или иной точки на карте.

Все это обеспечивает безопасность, удобство эксплуатации и комфорт, обеспечивающийся за счет упрощения и автоматизации управления.

Актуальной задачей является исследование мобильных компьютерных систем для проектирования и построения автоматизированных устройств.

По статистике, около 40% всех ДТП в зимнее время вызвано гололедом и снегопадом. Гололедица – это тонкий слой льда на поверхности земли, образующийся после оттепели или дождя в результате похолодания, а также замерзания мокрого снега и капель дождя. С целью фиксирования подобных событий и предупреждения водителя о возможном возникновении внештатной ситуации, оказывающей влияние на траекторию движения, необходима мобильная система, способная получить, запомнить и проанализировать данные об окружающей среде.

Для построения системы необходимо провести исследование датчика температуры DS18B20. Он обладает массой преимуществ по сравнению с другими устройствами

измерения температуры, одним из которых является очень маленькая погрешность при относительно большом диапазоне измерения температуры. Также исследуемый датчик совместим с большинством аппаратных вычислительных платформ, в том числе с контроллерами от фирм Arduino и Raspberry Pi.

Этот датчик также послужит для оптимизации работы других датчиков, например, датчика приближения, точность которого зависит от изменения температуры.

Построение математической модели мобильной системы

Датчик позволяет измерять температуру в диапазоне от -55°C до $+125^{\circ}\text{C}$. Сигнал, полученный с прибора, не нуждается в специальном преобразовании, так как уже является цифровым кодом замеренной температуры.

Как пояснялось ранее, погрешность замеров температуры для диапазона от -10°C до $+85^{\circ}\text{C}$ не превышает 0.5°C .

С целью исследования зависимости погрешности от температуры были проведены замеры при температурах в диапазоне от 0°C до 70°C с промежуточными расчетами ошибок измерений и произведено сравнение результатов с ожидаемыми ошибками соответствующих температур. Результаты исследований представлены в Таблице 1.

Таблица 1 Зависимость погрешности от температуры

t ($^{\circ}\text{C}$)	+3s Ошибка	Ожидаемая ошибка	-3s Ошибка
0	0.27	-0.15	-0.45
10	0.1	-0.18	-0.46
20	0.07	-0.2	-0.45
30	0.05	-0.18	-0.41
40	0.1	-0.14	-0.39
50	0.18	-0.07	-0.32
60	0.3	0.02	-0.26
70	0.45	0.15	-0.16

На рис. 1 приведена диаграмма зависимости погрешности температуры Δt ($^{\circ}\text{C}$) от температуры t ($^{\circ}\text{C}$) и проведена линия трендов.

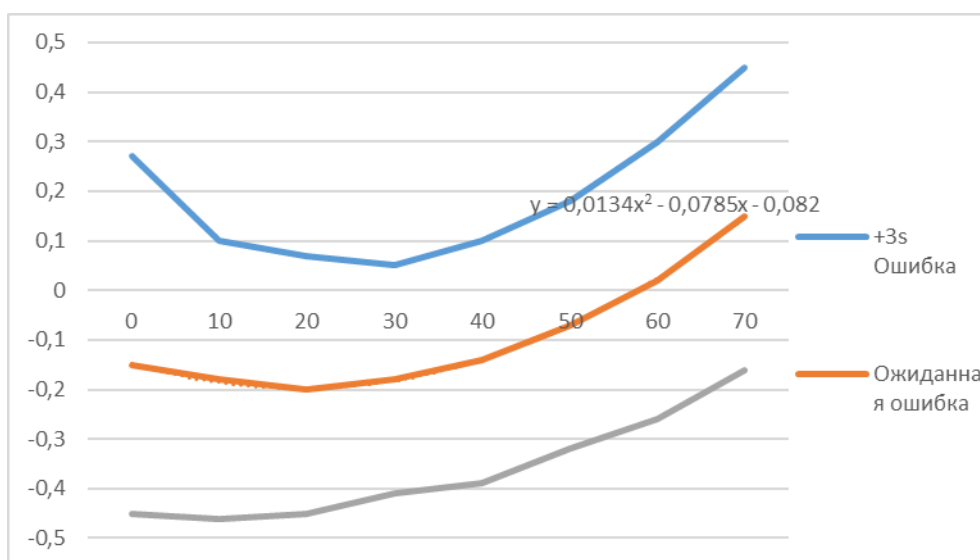


Рисунок 1 Зависимость погрешности от температуры

Используя аппроксимацию функции построенного графика получим формулу, описывающую зависимость погрешности температуры Δt (°C) от температуры t °C :

$$\Delta t = 0.0134 * t^2 - 0.0785 * t - 0.082.$$

Проверка адекватности модели

С целью проверки рассмотренной математической модели на адекватность были проведены исследования, заключающиеся в нахождении погрешности при сравнении расчетных показаний с экспериментальными. Результаты исследований представлены в Таблице 2.

Таблица 2 Расчётные и экспериментальные зависимости

trасч, °C	tэксп, °C	Δt
-10	-9.4	0.6
-9	-9.55	-0.55
-8	-7.46	0.54
-7	-6.5	0.5
-6	-6.49	-0.49
-5	-5.46	-0.46
-4	-3.57	0.43
-3	-2.6	0.4
-2	-1.61	0.39
-1	-1.4	-0.4
0	0.38	0.38
1	1.35	0.35
2	1.69	-0.31
3	2.7	-0.3
4	4.32	0.32
5	5.3	0.3
6	5.67	-0.33
7	7.29	0.29
8	8.28	0.28
9	8.69	-0.31
10	10.33	0.33

Исходя из погрешностей, полученных в ходе испытаний, произведена оценка адекватности математической модели, в соответствии с которой в модель мобильной системы внесены определенные характеристики и параметры, учитывающие неточности показаний датчика.

Объектный анализ мобильной системы с использованием UML

В работе [6] обоснован выбор объектно-ориентированного подхода (ООП) для построения программной модели интеллектуальной системы. Объектно-ориентированный подход опирается на такие понятия, как полиморфизм, наследование, абстракция и события. ООП обеспечивает гибкость архитектуры модели ввиду того, что вся система представляет собой набор совокупностей класс-объект. В модели мобильной системы могут быть как динамические, так и статические отношения между объектами, описание которых – одно из главных преимуществ объектно-ориентированного подхода.

Одним из методов ООП является графическое моделирование с помощью UML-диаграмм. Они позволяют наглядно представить систему в виде объектов и функционала для взаимодействия с ними и с самой системой. Для реализации этих целей самым подходящим вариантом является диаграмма использования (Use Case Diagram). Она включает в себя 2 типа сущностей: действующее лицо – актер и варианты использования – функциональная часть системы и взаимодействия с ней.

Составим UML-диаграмму, наглядно демонстрирующую взаимодействие с исследуемым устройством. UML-диаграмма продемонстрирована на рис. 2.

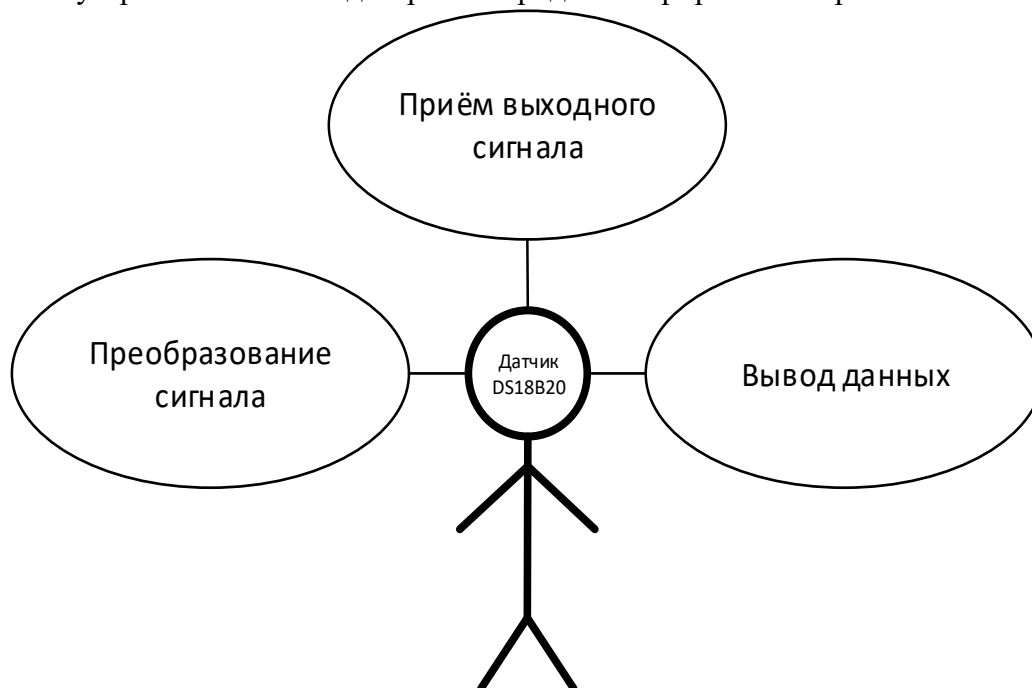


Рисунок 2 – Диаграмма взаимодействия датчика температуры DS18B20 с исследуемым устройством

В роли актера будет выступать датчик температуры DS18B20. В функциональную часть UML диаграммы входят методы:

- приём выходного сигнала;
- преобразование сигнала;
- вывод данных.

Разработка программно-имитационной модели

Для программной реализации моделирования работы исследуемого датчика выбран язык программирования C#. Данный язык программирования является объектно-ориентированным, относится к Си-подобным языкам, что несет за собой простоту написания и читаемости программного кода. Также C# поддерживает инкапсуляцию, наследование и полиморфизм.

Опираясь на то, что проверка работы датчика должна проводиться в мобильной системе, необходимо реализовать имитацию работы всей системы. Так как в системе может находиться неограниченное количество датчиков, схожих по параметрам или принципам работы, нет надобности создавать отдельный класс для каждого устройства. Необходимо лишь создать абстрактный класс, включающий параметры и методы, присущие всем датчикам. Затем, для реализации отдельных датчиков, достаточно наследовать классами датчиков ранее созданный абстрактный класс и добавлять поля и переопределять методы соответственно датчику, который реализует класс.

Программная реализация класса для имитации модели работы датчика DS18B20 на языке программирования C# представлена ниже:

```
using System;
using Sensor;

namespace DS18B20
{
    class DS18B20:Analog_Sensor
    {
        /// <param name="M">Единицы измерения</param>
        public string M = "°C";
        /// <param name="MI">Максимальное измеряемое значение</param>
        public int MI = 125;
        /// <param name="MX">Минимальное измеряемое значение</param>
        public int MX = -55;
        /// <param name="MO">Модель датчика</param>
        public string MO = "DS18B20";
        /// <param name="R">Частота опроса</param>
        public float R = 0.75F;
        public string Measurement_value { get; }
        public int Minimum_value { get; }
        public int Maximum_value { get; }
        public string Model { get; }
        public float Rate_of_change_of_values { get; set; }
        public DS18B20(string M, int MI, int MX, string MO, float R, bool P) : base(M,
MI, MX, MO, R, P)
        {
            Measurement_value = M;
            Minimum_value = MI;
            Maximum_value = MX;
            Model = MO;
            Rate_of_change_of_values = R;
        }
        public override float Mathematical_model(float T, float delta, bool P = true)
        {
            // t - заданная температура
            float t = T;
            // расчет фактической температуры
            return t = T + delta;
        }
    }
}
```

Класс DS18B20, представляющий имитацию работы датчика DS18B20 является наследником абстрактного класса Analog_Sensor. Он включает все необходимые для полного функционирования поля, свойства и методы. Наглядное представление класса представлено на диаграмме классов (рис. 3).

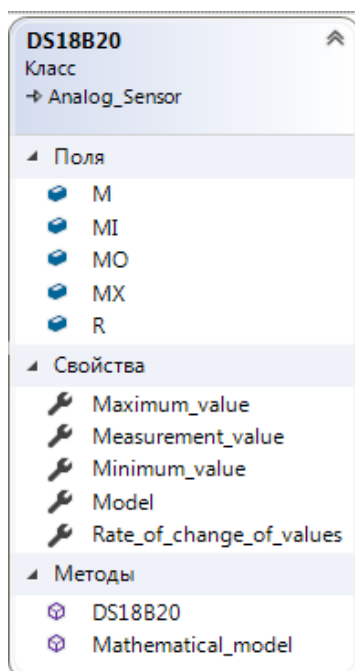


Рисунок 3 - Диаграмма классов

Класс содержит следующие поля:

- M – единицы измерения
- MI – максимальное измеряемое значение
- MX – минимальное измеряемое значение
- MO – модель датчика
- R – частота опроса датчика мобильной системой

Метод DS18B20 является конструктором класса и реализует инициацию параметров, общих с другими датчиками.

Метод `Mathematical_model` переопределяет унаследованный метод абстрактного класса `Analog_Sensor` и реализует имитацию работы датчика.

Выводы

Был проанализирован датчик температуры DS18B20. Были исследованы принципы его работы, основные характеристики и преимущества. Исследована зависимость погрешности от температуры, а также разработана программно-имитационная модель датчика.

Литература

1. Иго, Том Arduino, датчики и сети для связи устройств / Том Иго. - М.: БХВ-Петербург, 2011. 822 с.
2. Саченко, А.А. Измерение температуры датчиками со встроенными калибраторами / А.А. Саченко. - М.: Книга по Требованию, 2012. 456 с.
3. Преображенский, В. П. Теплотехнические измерения и приборы. - М.: Энергия, 1978. 704 с.
4. Чистяков, С. Ф., Радун, Д. В. Теплотехнические измерения и приборы. - М.: Высшая школа, 1972. 392 с.
5. Страуструп, Б. Язык программирования C++. М.: Радио и связь, 2013. 350 с.
6. Николаенко, Д. В., Плешкова О. А. Объектная модель интеллектуальной системы управления транспортом // Современные тенденции развития и перспективы внедрения инновационных технологий в машиностроении, образовании и экономике: материалы II Международной научно-практической конференции. – Ростов–на–Дону: ДГТУ, С. 10–14.

СЕКЦИЯ 9

СИСТЕМНЫЙ АНАЛИЗ И
УПРАВЛЕНИЕ.
ПРИКЛАДНАЯ ИНФОРМАТИКА

ИУС
МКМ 2019

УДК 004.588

РАЗРАБОТКА СТРУКТУРЫ ВИРТУАЛЬНОЙ ЛАБОРАТОРИИ ПО ВЫЧИСЛИТЕЛЬНОЙ ГЕОМЕТРИИ

К.А. Бурзак, Е.Н. Едемская

Донецкий национальный технический университет, г. Донецк
кафедра искусственного интеллекта и системного анализа

E-mail: karina_tsukanova@inbox.ru

Аннотация:

Бурзак К.А., Едемская Е.Н. Разработка структуры виртуальной лаборатории по вычислительной геометрии. Рассмотрены виды и приемы дидактических работ в обучающей деятельности. Приведен тематический перечень разделов вычислительной геометрии и проанализирован теоретический материал по данной дисциплине. Разработаны структурная схема виртуальной лаборатории и сценарий курса обучения.

Annotation:

Burzak K.A., Edemskaya E.N. The types and techniques of didactic works in teaching activities are considered. A thematic list of sections of computational geometry is given and theoretical material on this discipline is analyzed. Developed a block diagram of a virtual laboratory and a training course scenario.

Общая постановка проблемы обучения вычислительной геометрии

В обучении специалистов широкого профиля предусматривается углубление теоретической базы, овладение фундаментальными основами инженерной и управленческой деятельности, существенное улучшение практической подготовки.

Одним из перспективных способов повышения эффективности процесса обучения является его автоматизация, т.е. использование в качестве средства обучения современной вычислительной техники.

Основным стимулом развития вычислительной геометрии как дисциплины был прогресс в компьютерной графике и системах автоматизированного проектирования, однако многие задачи вычислительной геометрии является классическим по своей природе, и могут появляться при математической визуализации.

При изучении алгоритмов решения геометрических задач главной проблемой является, как правило, недостаточно развитое пространственное мышление обучаемого. Это влечет за собой непонимание основных принципов построения как простых, так и сложных проекций, в том числе и всевозможных чертежей. Школьный курс черчения, который является базовым при изучении инженерной графики, на данный момент во многих школах просто не изучается или изучается в сокращенном объеме, что создает трудности при изучении данной дисциплины в рамках учебного плана вуза. Самостоятельное же обучение с использованием литературы зачастую не дает желаемых результатов, поскольку: во-первых – требует умения самостоятельно работать, во-вторых – занимает больше времени, нежели обучение с учителем.

Выходом в этой ситуации может быть обучающая система или виртуальная лаборатория, которая предоставит обучаемому необходимый теоретический материал и задачи, снабженные анимированными представлениями наиболее сложных в понимании чертежей.

Перед реализацией виртуальной лаборатории по любому из учебных курсов необходимо выполнить следующее:

- определить тематическое содержание обучающего материала;
- выделить набор базовых тем для тестирования;
- разработать структуру лаборатории;
- разработать сценарий курса обучения.

Цель статьи – разработка структуры виртуальной лаборатории для решения задач вычислительной геометрии, а также сценария курса обучения.

Разработка структурной схемы виртуальной лаборатории

С точки зрения педагогики можно выделить три уровня обучения:

- понимание;
- усвоение;
- применение.

На каждом из этих уровней решаются свои дидактические задачи, действуют свои закономерности. Виды и приемы работ могут быть сгруппированы в 12 подсистем (см. рис. 1).

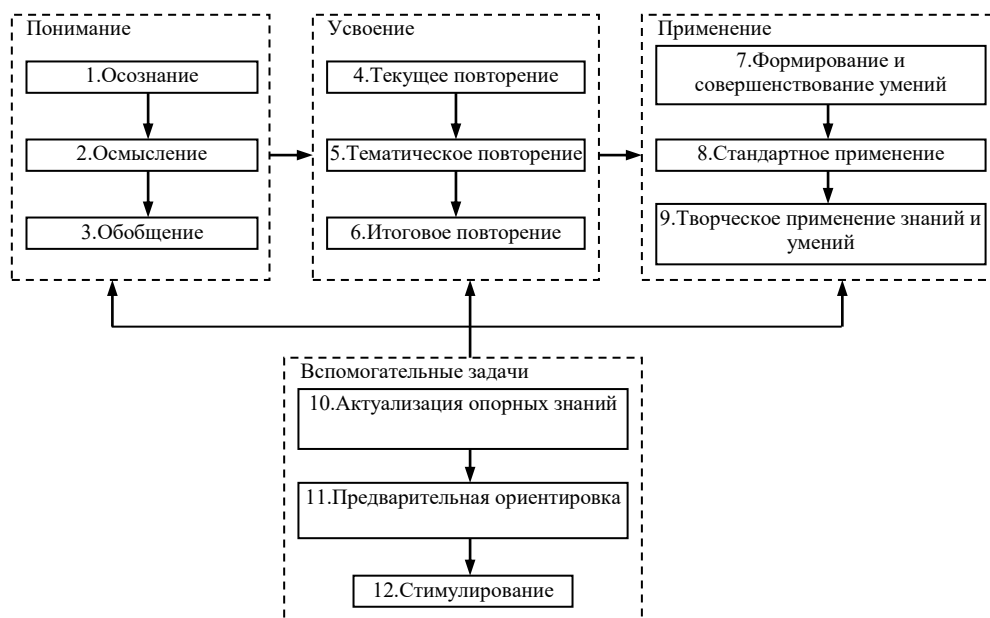


Рис.1. Виды и приемы дидактических работ

Каждый из видов работ может иметь различное представление в электронной лаборатории, однако, для лучшего усвоения материала необходимо наличие всех видов. Их комбинация определяет сценарий обучения.

Прежде чем излагать материал, необходимо проверить уровень знаний обучаемого. В зависимости от этого система должна определить набор знаний, представляемых в уроках.

Для того чтобы приступить к изучению вычислительной геометрии достаточно знаний элементарной геометрии на уровне понятий «точка», «прямая», «плоскость», а также основные аксиомы и теоремы, связанные с этими понятиями.

Каждый урок содержит теоретический материал и набор задач. Поскольку навыки пространственного мышления у всех развиты в разной степени, и определить их автоматически не представляется возможным, то анимированное представление процесса построения чертежей должно вызываться по желанию обучаемого. Каждый урок заканчивается тестированием. В зависимости от результатов обучаемый либо получает возможность перейти к следующему уроку, либо возвращается на начало текущего урока.

Тестовые вопросы должны содержать как практический, так и теоретический материал. Для того чтобы избежать ошибок в интерпретации ответа обучаемого необходимо использовать систему тестов с указанием вариантов ответов. Выбор этой системы тестирования обусловлен также тем, что некоторые вопросы предполагают развернутые ответы в текстовом виде или рисунки.

Для большей достоверности результатов тестирования необходимо:

- использовать не менее трех вариантов ответов;
- включить в тест вопросы, различные по формулировке, но одинаковые по смыслу

(контрольные вопросы).

Кроме режима обучения по заданному сценарию, система должна предоставлять возможность графического решения стандартных задач вычислительной геометрии.

Структура виртуальной лаборатории представлена на рис. 2.

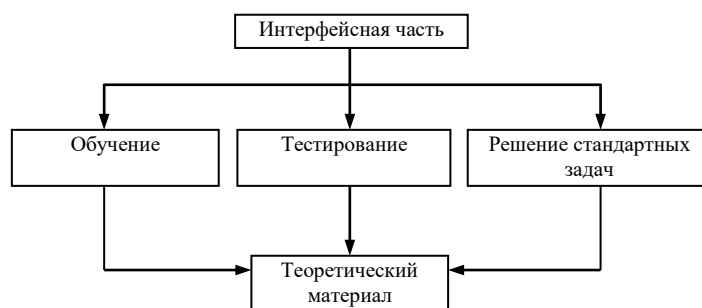


Рис.2. Структура виртуальной лаборатории

Как видно из схемы, лаборатория должна содержать три основных блока:

- 1) обучение;
- 2) тестирование;
- 3) решение стандартных задач.

Блок обучения должен включать в себя предоставление теоретического материала, сопровождаемого статичными чертежами, демонстрирующими понятия и методы вычислительной геометрии.

Блок тестирования должен предоставлять пользователю (обучаемому) тестовые вопросы с вариантами ответов по интересующей теме для проверки текущего уровня знаний. После тестирования обучаемому должен выводиться результат тестирования.

Блок решения стандартных задач должен позволять строить некоторые чертежи по исходным данным, задаваемым пользователем.

Из рисунка, что данные блоки связаны с теоретическим материалом, т.е. должна присутствовать возможность получения теоретической справки в любой момент работы с лабораторией.

Разработка структуры курса обучения

Вычислительная геометрия – это раздел информатики, изучающий алгоритмы решения геометрических задач.

В ней рассматриваются такие задачи как:

- 1) проецирование точки;
- 2) проецирование прямой линии;
- 3) проецирование плоскостей;
- 4) взаимное положение плоскостей;
- 5) взаимное положение прямой и плоскости;
- 6) перпендикулярность геометрических объектов;
- 7) метод перемены плоскостей проекций;
- 8) плоскопараллельное перемещение;

- 9) гранные поверхности;
- 10) развертка гранной поверхности;
- 11) кривые линии и поверхности;
- 12) пресечение поверхностей с плоскостями;
- 13) сечение тел и развертка;
- 14) пересечение прямой и поверхности;
- 15) аксонометрия.

В результате анализа перечня разделов вычислительной геометрии и изучения теоретического материала была разработана тематическая структура курса обучения, представленная на рисунке 3.

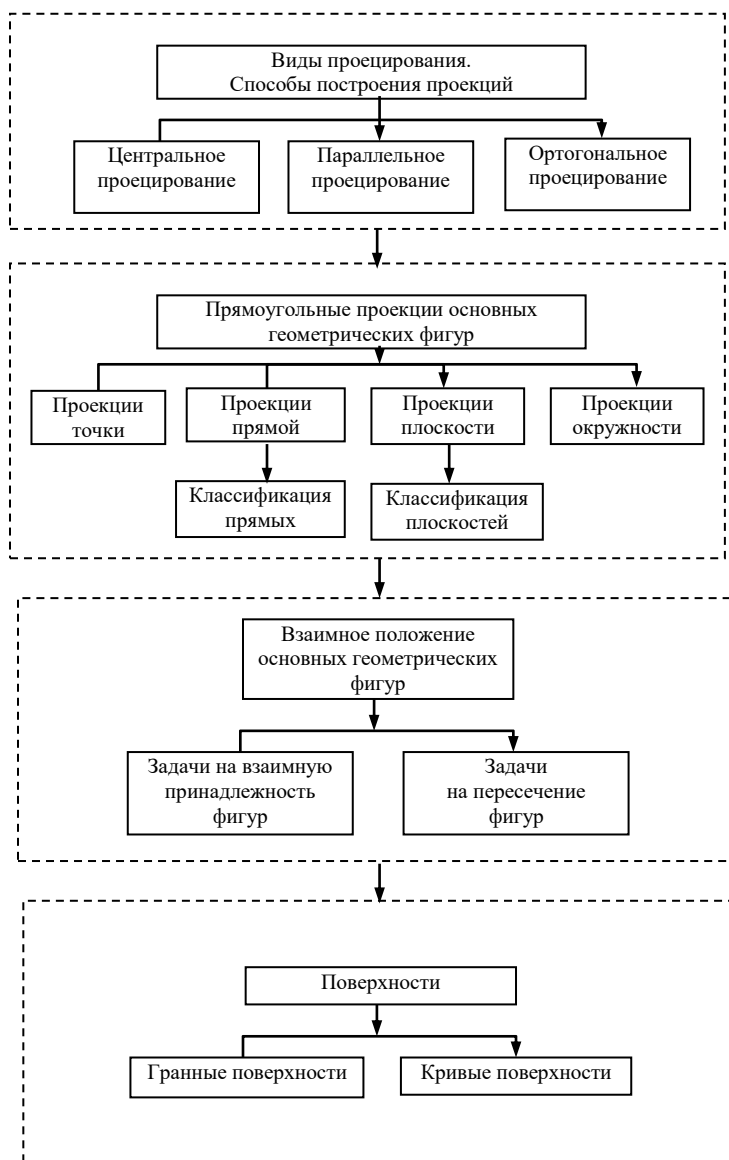


Рис. 3. Тематическая структура курса

Курс имеет иерархическую структуру, так как изучение какого-либо материала подразумевает наличие знаний по предшествующим разделам или темам.

Каждая из групп блоков, обведенных пунктирной линией объединяет в себе тематический набор, которые пользователю желательно освоить перед началом изучения следующей группы.

Таким образом, сценарий курса имеет вид «обучение+тестирование». Обучение представляет собой подачу пользователю текста с иллюстративным или (и) анимационным материалом с возможностью выбора интересующего раздела. Однако при произвольном обращении к разделам обучаемому желательно пройти тестирование по материалам предшествующих тем.

Выводы

Главное назначение обучения (а, следовательно, и обучающих программ) – овладение умениями. Исходя из этого, одной из основных функций учебной программы является управление познавательной деятельностью обучаемого. Механизмом осуществления такой деятельности является решение задач.

Таким образом, структура обучающая система должна состоять из 3 частей:

- 1) теоретической;
- 2) тренирующей;
- 3) контролирующей.

Одна из наиболее существенных особенностей проектирования обучающих программ состоит в том, что основу таких программ составляют сценарии курса обучения, включающие правила перехода от одной ситуации к другой в зависимости от действий обучаемого.

Главное назначение виртуальной лаборатории по вычислительной геометрии является базовая общеинженерная подготовка: развитие пространственного представления и воображения, конструктивно-геометрического мышления, выработка знаний и навыков, необходимых для построения чертежей.

Следовательно, лаборатория должна включать в себя следующие компоненты:

- теоретический материал по основным разделам вычислительной геометрии;
- примеры решения наиболее распространенных задач;
- средства контроля (тестирования) знаний обучаемого.

Литература

1. Трухин, А.В. Об использовании виртуальных лабораторий в образовании / А.В. Трухин // Открытое и дистанционное образование. Томск, 2002. №4 (8). С.67.
2. Столбунская, А.С. Автоматизация процесса обучения при помощи виртуальных лабораторий [Текст] / А.С. Столбунская, И.Д. Паламарчук, Д.М. Бочаров // Информатика, управляющие системы, математическое и компьютерное моделирование в рамках III Международного Научного форума Донецкой Народной Республики (ИУСМКМ-2017): VIII Международная научно-техническая конференция, 25 мая 2017 – Донецк: ДонНТУ, 2017. – С. 406-409.
3. Виртуальные лаборатории [Электронный ресурс] / Интернет-ресурс. – Режим доступа : [www/ URL: http://kpfu.ru/docs/F324157708/Virtualnye_laboratorii.pdf](http://kpfu.ru/docs/F324157708/Virtualnye_laboratorii.pdf). – Загл. с экрана.
4. Технологии обучения. Виртуальные лаборатории для дистанционного обучения, вузов и школ [Электронный ресурс] / Интернет-ресурс. – Режим доступа : <http://zillion.net/ru/blog/247/virtual-nyie-laboratorii-dlia-distantionnogo-obucheniia-vuzov-i-shkol>. – Загл. с экрана.
5. Виртуальная лаборатория VirtuЛаб [Электронный ресурс] / Интернет-ресурс. – Режим доступа : [www/ URL: http://www.virtulab.net/](http://www.virtulab.net/). – Загл. с экрана.
6. Ошибки проектирования виртуальной среды дистанционного обучения [Электронный ресурс] / Интернет-ресурс. – Режим доступа : [www/ URL: http://www.web-learn.ru/biblioteka-online/50-error-designing-of-the-virtual-elearning-environment](http://www.web-learn.ru/biblioteka-online/50-error-designing-of-the-virtual-elearning-environment). – Загл. с экрана.

УДК 004.9

АНАЛИЗ ВНЕДРЕНИЯ СИСТЕМЫ ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА УНИВЕРСИТЕТА С ИСПОЛЬЗОВАНИЕМ МЕТОДОЛОГИИ ФУНКЦИОНАЛЬНОГО МОДЕЛИРОВАНИЯ IDEF0

Бутенко Д.А., Радевич Е.В.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: factorofzero@gmail.com

Аннотация:

Бутенко Д.А., Радевич Е.В. Анализ внедрения системы электронного документооборота (далее СЭД) университета с использованием методологии функционального моделирования IDEF0. Предложена методика подготовки к внедрению СЭД в университете. Проведен анализ основных бизнес-процессов, выявлены основополагающие проблемы в системе документооборота и рассмотрены пути их решения. Полученная методика позволит сократить временные затраты на выполнение основных технологических операций, снизить влияние ошибок, которые возникают вследствие влияния человеческого фактора.

Butenko DA, Radevich E.V. Analysis of the implementation of the university's electronic document management system using the IDEF0 functional modeling methodology. The method of preparation for the introduction of EDS at the university is proposed. The analysis of the main business processes, identified the underlying problems in the workflow system and considered ways to solve them. The resulting technique will reduce the time spent on the implementation of basic technological operations, reduce the impact of errors that occur due to the influence of the human factor.

Общая постановка проблемы.

Для поддержки и развития научной деятельности ВУЗа необходимо применять актуальные методы и современные технологии для ее управления. Информационно-аналитическая система поддержки научной деятельности, входящая в корпоративную информационную среду ВУЗа, является наиболее совершенным средством управления, поскольку способна оперировать огромными массивами информации, связанными с объектом управления, интегрировать информационно-аналитическую поддержку научной деятельности с поддержкой других видов деятельности университета. Для подготовки методики создания информационного и программного обеспечения информационно-аналитической системы на основе анализа показателей научной деятельности необходимо структурировать и оптимизировать совокупность моделей, позволяющих создавать корректные компоненты всех видов обеспечения системы: аналитические и графовые модели, а также модели «сущность-связь» и функциональные модели в формате IDEF0.

Анализ разработок и публикаций.

Литературный обзор показал отсутствие существующих методик подготовки к внедрению СЭД в университете. Методика была разработана на основе опыта внедрения СЭД в университеты РФ, аккумулируя и дополняя знания по внедрению и оптимизации СЭД докторов наук (Ю.Ю. Петров [1], В.Г. Лим [2]), кандидатов наук (Ю.В. Химич [3], С.Г. Клименко [4], А.П. Баринов [5]). Разработанная методика позволяет выявить и проанализировать существующие проблемы в системе документооборота и увеличить степень готовности высшего учебного заведения к внедрению СЭД.

Формулировка цели статьи.

Охарактеризовать процесс внедрения системы электронного документооборота в университете на основе существующих методик с применением методологии функционального моделирования IDEF0.

Изложение основного материала исследования. На подготовительном этапе были проанализированы особенности деятельности каждого подразделения вуза и определен конечный набор документов, который функционирует внутри каждого подразделения и однозначно определен в утвержденной номенклатуре дел.

Результирующие информационные потоки представляют собой данные об отработке документов, а также отчетность по данным об отработанных документах. На рисунке 1 показана диаграмма декомпозиции основного бизнес-процесса.

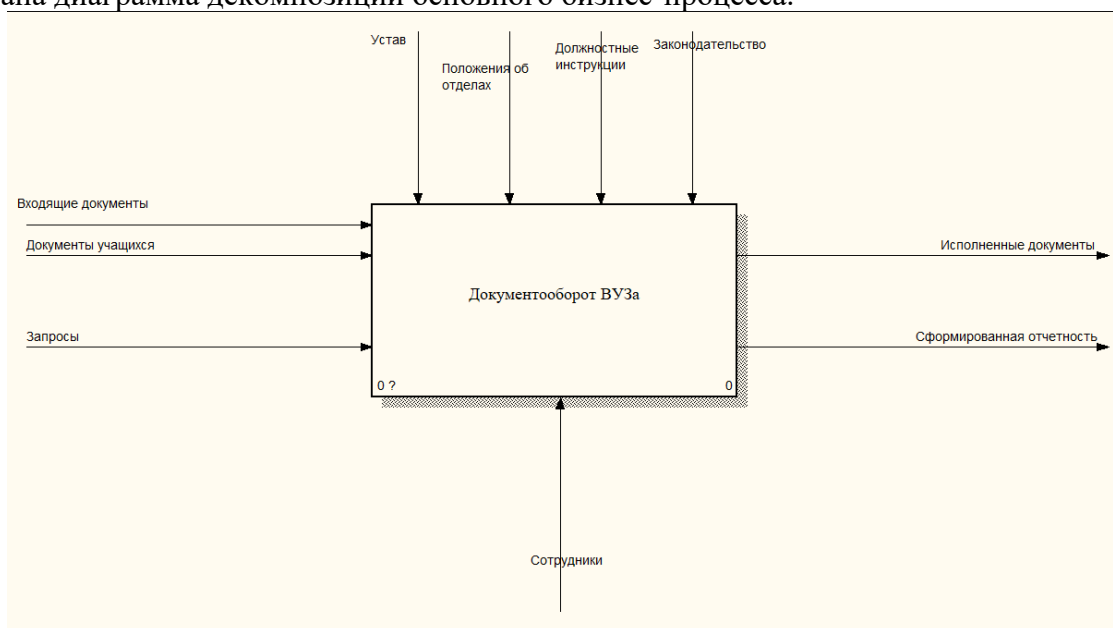


Рисунок 1 – Контекстная диаграмма

Основной бизнес-процесс в технологии документооборота университета включает в себя: ведение документации учащихся, учет входящих документов, формирование приказов, формирование отчетности (рис.2-4):

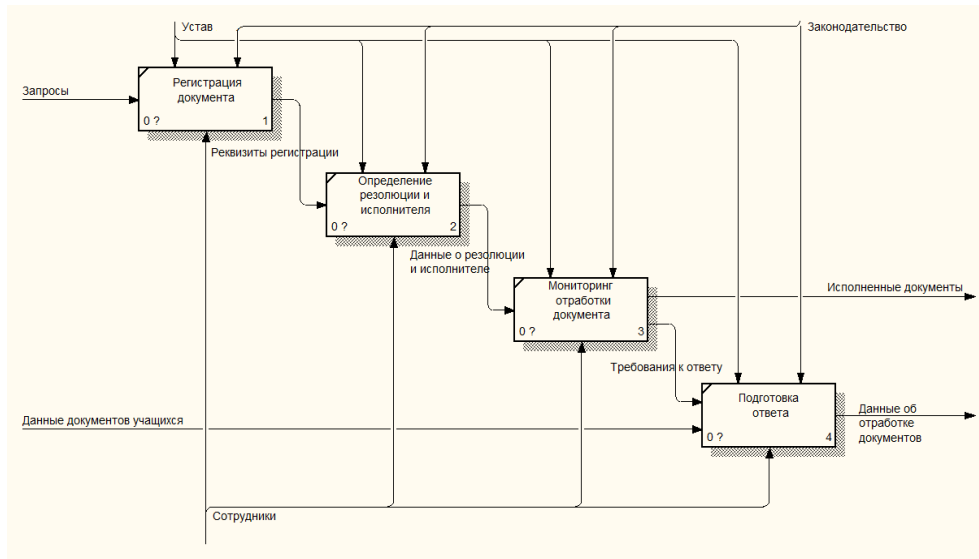


Рисунок 2 - Диаграмма декомпозиции процесса ведения документации

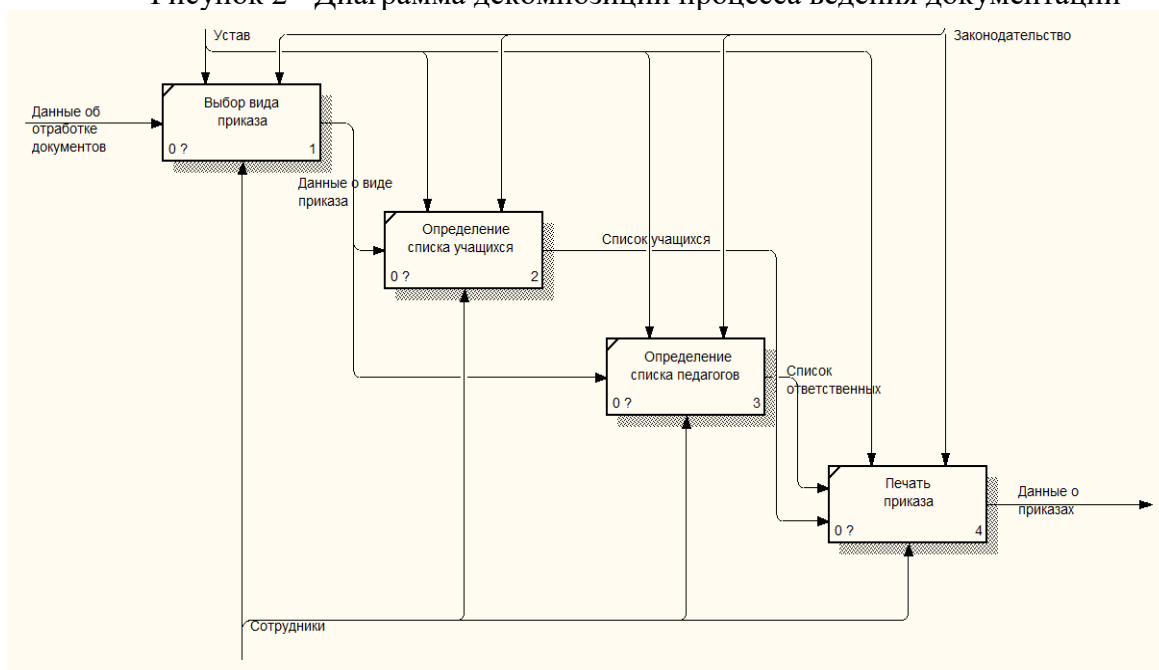


Рисунок 3 - Диаграмма декомпозиции отработки документов

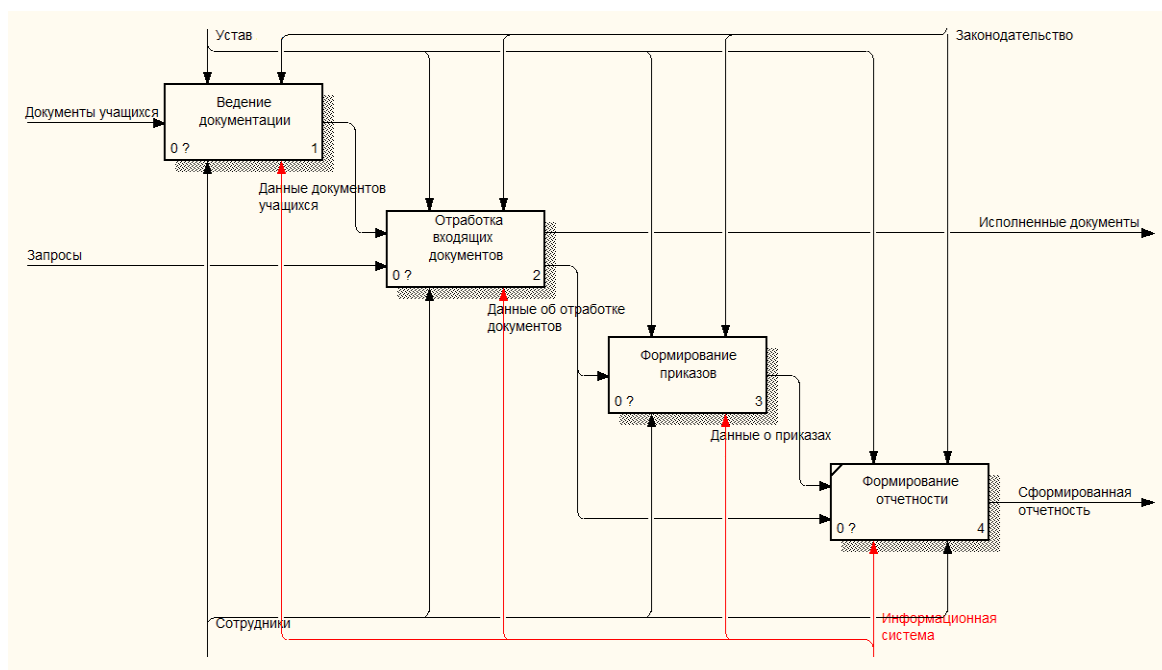


Рисунок 4 - Диаграмма декомпозиции формирования приказов

Анализ существующей технологии документооборота позволил выявить следующие недостатки:

- отсутствует структурированная информация о входящих документах;
- отсутствует возможность формирования сводных показателей по различным параметрам, что не дает возможности проведения анализа эффективности отработки документов, отсутствует возможность формирования сводных данных в разрезе исполнителей по исполненным и просроченным документам, что приводит к падению дисциплины в процессе их отработки;
- отсутствует возможность формирования справок с места учебы в автоматизированном режиме;
- отсутствует возможность автоматического формирования приказов, что приводит к необходимости поиска в личных делах данных об учащихся и педагогах, включенных в приказ, что затратно по времени и предполагает возможности ошибок, связанных с человеческим фактором.

Внедрение информационной системы в технологию документооборота университета позволит сократить временные затраты на выполнение основных технологических операций, снизить влияние ошибок, возникающих вследствие влияния человеческого фактора.

Исходя из рисунка 5, для каждого из бизнес-процессов предполагается внедрение информационной системы.

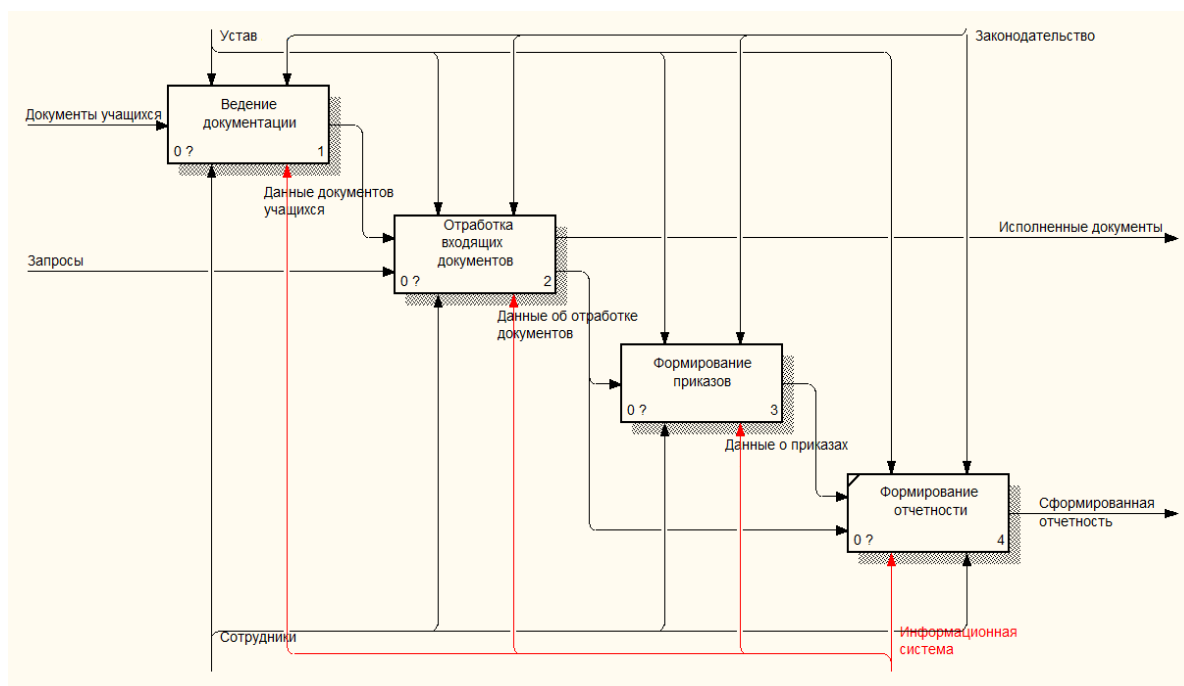


Рисунок 5 – Диаграмма декомпозиции основного бизнес-процесса

На втором этапе необходимо приобрести СЭД, и здесь возникают проблемы, связанные со сложной схемой финансирования внедрения проекта (требования по проведению конкурсов, сложная схема принятия решений и прочее). На ранних этапах это вызывает особые сложности. Работу по внедрению СЭД в государственных учреждениях необходимо начинать с небольшого пилотного проекта, задача которого стоит в выявлении эффекта от внедрения системы и выработки базиса для составления проекта реальной системы [6].

После выбранной экспериментальной площадки происходит настройка необходимого программного обеспечения: серверной части, клиентской части, настройки рабочих мест и подключения необходимого дополнительного оборудования.

В процессе настройки происходит инструктаж сотрудников и кратковременное обучение возможностям системы, при необходимости ранее обученная группа консультантов поможет первых шагах сотрудникам.

Следующим шагом этапа является уточнение и занесение в систему документов, заявленных в номенклатуре дел структурных подразделений.

После настройки рабочих мест и подготовительной работы с сотрудниками начинается непосредственная работа в СЭД. Сроки данного этапа должны включать полный цикл работы подразделений, и для большинства структур университета им является учебный год.

На данном этапе ставится несколько задач:

- выявление целесообразности применения СЭД;
- удобство использования выбранной СЭД;
- накапливание проблемных ситуаций и выработка оптимальных их решений;
- выявление узких и слабых мест системы и варианты их преодоления;
- обучение персонала подразделений СЭД;
- отработка маршрутов документов;

Результатом этапа будет накопленный опыт применения системы электронного документооборота в документообороте вуза.

После реализации указанных этапов университет будет готов к массовому внедрению СЭД.

Выводы.

Предлагаемый подход позволит не только оптимально использовать новейшие информационные технологии для организации процессов управления документами, но и максимально эффективно использовать СЭД, включив ее в качестве центрального звена в информационную среду, оптимизировать работу с классификаторами и справочниками, стандартизировать процессы обработки документов, создавать единые сквозные процессы управления документами в университете.

Литература

1. Петров Ю.Ю. Методы управления промышленным предприятием на основе электронного документооборота. / Санкт-Петербург: ИНЖЕКОН, 2003 г., 204с.
2. Лим В.Г. Разработка информационно-аналитических средств безбумажного документооборота в строительном производстве. / Москва: ЗАО «ЦНИИОМТП», 2005 г., 331с.
3. Химич Ю.В. СЭД (методика внедрения). / Москва: Портал «TAdviser», 2010 г. [Электронный ресурс] URL: <http://www.tadviser.ru/index.php/Статьи:СЭД> (методика внедрения).
4. Клименко С.Г. Разработка моделей и методов взаимодействия интернет-ориентированных систем управления документооборота со средствами аутентификации. / Воронеж: Воронежский экономико-правовой институт, 2011 г., 157с.
5. Баринев А.П. Разработка системы информационной поддержки контроля производственных процессов на базе терм-связности элементов электронного документооборота. / Москва: МАДИ, 2008 г., 158с.
6. Андрейченко, А.А. Технология реализации системы электронного документооборота вуза на основе объектно-ориентированной программной среды // “Новые информационные технологии”. Тезисы докладов XVII Международной студенческой конференции-школы-семинара – М.: МИЭМ, 2009 – 399 с.

УДК 004.891.2

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ СИСТЕМЫ НЕЧЕТКИХ ПРАВИЛ ДЛЯ АВТОМАТИЗАЦИИ ОТДЕЛЬНЫХ УЗЛОВ МЕТАЛЛУРГИЧЕСКОГО ПРОИЗВОДСТВА

Гуляев В.А., Орлов Ю.К.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: gvard12@gmail.com

Аннотация:

Гуляев В.А., Орлов Ю.К. Программная реализация системы нечетких правил для автоматизации отдельных узлов металлургического производства. Рассмотрены основания для применения теории нечетких множеств в металлургическом производстве. Разработана концепция программного решения. Разработан прототип программного обеспечения, реализующего процедуру создания и работы системы нечетких правил.

Annotation:

Gulyaev V.A., Orlov J.C. Software implementation of fuzzy logic system for the automation of particular parts of metallurgical production. The grounds for fuzzy set theory application in metallurgical production was considered. The concept of a software solution has been developed. Software prototype that implements the procedure of creating and operating a system of fuzzy rules has been developed.

Постановка проблемы

Автоматизация металлургического производства является одним из важнейших направлений деятельности предприятий этой отрасли на постсоветском пространстве. Уровень механизации, который был достигнут еще в советские годы, позволял производить продукцию в большом объеме, однако недостаточно конкурентоспособного качества с точки зрения мирового рынка, что выражалось в низком спросе на неё. Переход от механизации к автоматизации производился преимущественно на больших предприятиях стратегического значения, где поддержание эффективной и бесперебойной деятельности было первоочередным заданием руководства. С распространением компьютеров и их применении в промышленности возникла возможность наладить более безопасное и надежное производство путём внедрения компьютеризированных автоматизированных систем управления технологическими процессами (АСУТП). В то же время, появилась возможность использования программных реализаций сложных процессов управления таких, как ПИД-регулирование, ШИМ-модуляция и др., а также внедрения ранее либо в принципе не испытанных, либо мало апробированных математических методов. К таким методам следует отнести теорию игр, нечеткое управление и многие другие.

Возможность применения системы нечетких правил в металлургической отрасли обуславливается тем, что некоторые технологические процессы не являются четко нормируемыми и решения об изменении тех или иных параметров человек, который отслеживает и управляет процессом, принимает не только исходя из понимания физических и химических процессов, но и своего опыта. То есть, иногда нельзя только задать четкую аварийную границу по какому-либо параметру и надеяться на то, что, дойдя до него и произведя отключение оборудования, не будет нанесен вред имуществу или людям. В то же

время, чаще всего сложно достоверно оценить подобные границы, поэтому технолог – человек, ответственный за технологический процесс – мог бы оперировать некоторыми интервалами. К тому же, бывают случаи, когда непредвиденное изменение значений нескольких параметров может привести к аварии при том, что по отдельности они не доходят до аварийных и даже регламентных границ.

Пример возможного применения нечеткой логики в металлургии

В качестве примера процесса, в который могла бы быть внедрена нечеткая логика, можно привести процесс аварийной остановки компрессора. Причинами аварийной остановки по измеряемым значениям компрессора чаще всего могут быть:

- повышенная вибрация ступеней или подшипников;
- повышенное давление или температура ступеней;
- низкий проток воды на охлаждение.

В зависимости от конструкции компрессора могут варьироваться количество измеряемых величин, поэтому далее речь будет идти обобщенно по ним.

Для начала необходимо определить списки входных и выходных переменных. На рисунке 1 показана схема этих переменных с указанием возможных состояний.

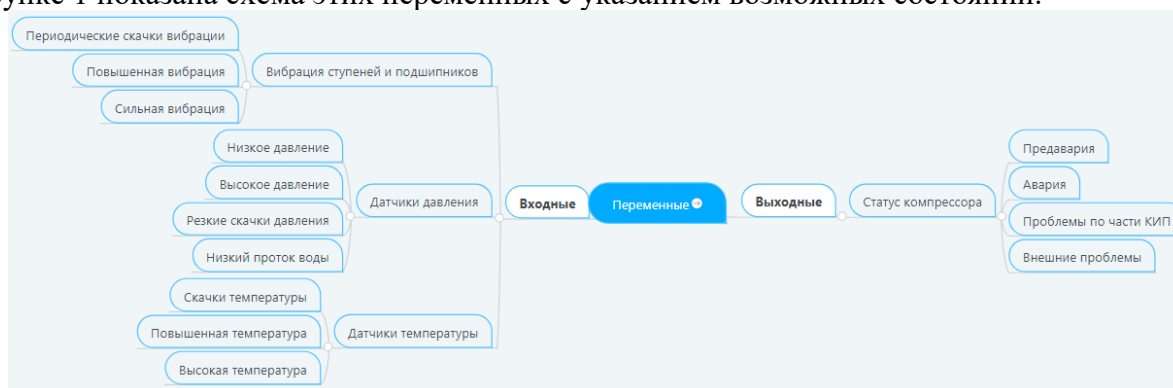


Рис. 1. Схема переменных

В этой схеме на первом уровне отображено деление на входные/выходные переменные, на втором – на лингвистические переменные, на третьем – на состояния лингвистических переменных.

Следует отметить, что лингвистические переменные «Датчики давления» и «Датчики температуры» являются намеренным упрощением, которое демонстрирует общность большинства параметров, входящих в эти группы. На практике применение таких групп параметров может быть обусловлено выделением в некоторые паттерны для того, чтобы, привязав значение одного из датчиков к паттерну, использовать уже существующие правила без необходимости внесения коррекций в уже налаженные механизмы преобразования. Необходимо учесть, что в таком случае значения, описывающие состояния датчиков в паттерне, распространятся и на новые подвязанные датчики. Также следует отметить, что в практике АСУТП состояния тех или иных установок и их параметров сводятся к двум значениям – «Авария» и «Предавария» [2], что чаще всего обусловлено ограничениями SCADA-систем. Предложенная схема позволяет расширить перечень для компрессора по крайней мере еще на два: «Проблемы по части КИП» и «Внешние проблемы», т.е. по требованию список состояний может расширяться.

Далее приведены примеры нечетких правил, которые могли бы быть использованы при построении системы автоматизации контроля и управления параметрами компрессора:

- Если *вибрация повышена* или *повышена температура*, то статус компрессора – *предавария*;

- Если *низкое давление* или *высокое давление* или *высокая температура* или *сильная вибрация*, то статус компрессора – *авария*;
- Если *низкое давление* и *периодические скачки вибрации* или *низкий проток воды*, то статус компрессора – *внешние проблемы*;
- Если *периодические скачки вибрации* или *резкие скачки давления* или *скачки температуры* или (*низкое давление* и *высокое давление*), то статус компрессора – *проблемы по части КИП*.

Описанные правила являются тривиальными, но тем не менее даже они могут быть использованы, чтобы гипотетическая система смогла работать в режиме рекомендаций без обратной связи с управлением. Также в представленных примерах сознательно опущено присвоение значений и представлений состояниям лингвистических переменных, т.к. они могут варьироваться в зависимости от характеристик оборудования.

Концепция программного решения

Прежде чем приступить к формированию концепции программы необходимо определиться с тем, как, собственно, будет работать программа. На рисунке 2 показана блок-схема работы системы нечетких правил.

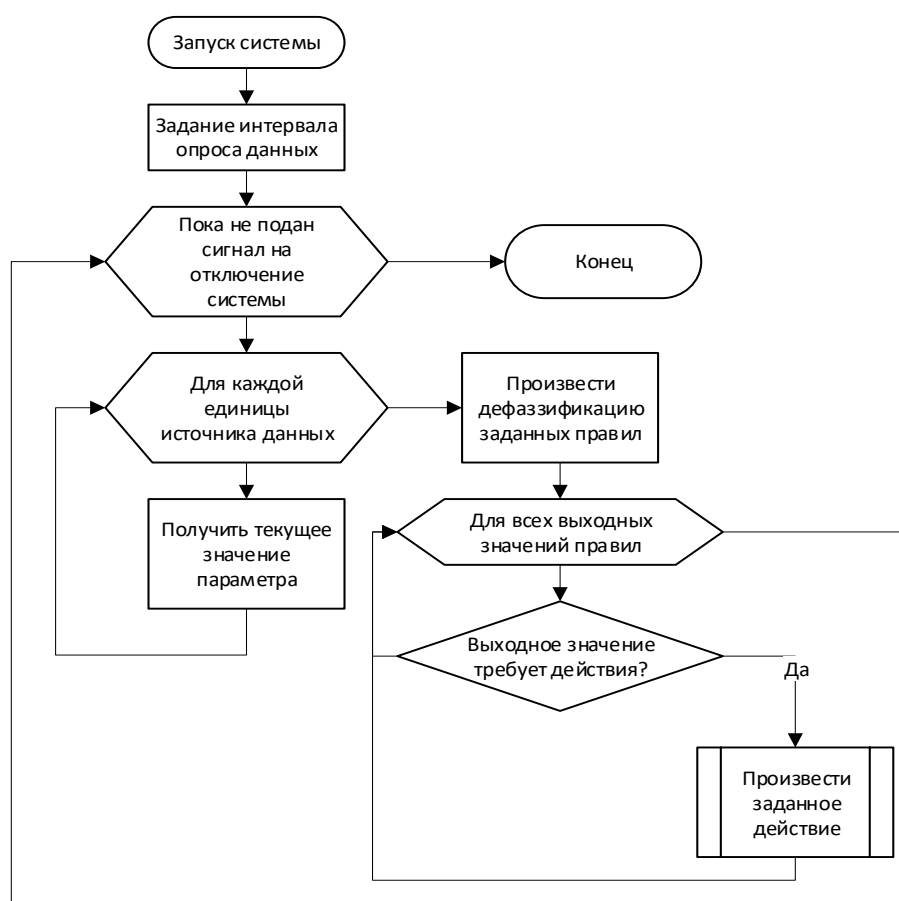


Рис. 2. Блок-схема работы системы

Задание самих правил для работы системы удобно реализовать в виде, сходном с тем, который используется в пакете Fuzzy Logic Toolbox программы MATLAB.

Учитывая специфику среды, в которой работает система, список источников данных откуда она может получать информацию может значительно варьироваться в зависимости от

технического исполнения среднего и верхнего уровней АСУТП. В частности, на среднем уровне АСУТП, где представлены программируемые логические контроллеры (далее – ПЛК), могут варьироваться модели этих ПЛК. К наиболее распространенным ПЛК можно отнести контроллеры компаний Siemens (S7), Rockwell (Allen-Bradley) и Mitsubishi (Alpha, Q) [3]. На верхнем уровне АСУТП вариация источников данных значительно возрастает, поскольку здесь могут использоваться различные SCADA-системы (проприетарные Siemens WinCC, Rockwell RSView32 и др. или сторонние, как, например, Wonderware InTouch), но тут возможно сведение к некоторым общностям – большинство этих систем могут использовать SQL базы данных для накопления получаемых данных и их отображения в виде графиков.

Таким образом, для получения входных данных для системы необходимо использование как драйверов или библиотек для подключения системы к ПЛК, так и методы для подключения и извлечения данных из SQL баз данных.

На выходе системы, как видно из рисунка 1, могут быть параметры, обозначающие те или иные состояния оборудования для оператора, но также, в перспективе, на выходе из системы могут быть значения для, например, регуляторов. Однако, для реализации непосредственного воздействия на параметры технологического процесса необходимы значительные исследования в области определения интервалов выходных элементов нечетких множеств для минимизации риска нанести вред оборудованию или персоналу. К тому же, не менее важно отсутствие коллизий с уже реализованными в ПЛК алгоритмами, когда загруженная в память программа задает одно значение, а система отправляет на ПЛК другое. Поэтому, в качестве тестового режима и, собственно, для определения этих интервалов и коллизий, система должна некоторое время проработать в условиях отсутствия доступа к выходным переменным для оценки корректности заданных множеств. Таким образом, стоит вопрос того, как организовать формирование выходных данных в виде, пригодном для анализа.

Для программной реализации рассматриваемой системы был выбран язык программирования C#, поскольку IDE для него (Visual Studio) представляет удобные инструменты для создания графического интерфейса пользователя, необходимый для формирования нечетких правил, а также на нем написаны библиотеки, которые необходимы для создания системы [4]. Интерфейс системы будет реализован при помощи WPF. Для реализации системы нечетких множеств, создания правил и процедур фаззификации и дефаззификации была выбрана библиотека Accord.Fuzzy как наиболее используемая по данным сайта github.com (выводы сделаны на основании количества поставленных звезд). Для подключения к ПЛК компании Siemens используется библиотека S7.Net, к ПЛК компании Rockwell – Corsinvest.AllenBradley. Для подключения к SQL базам данных (а именно Microsoft SQL Server, поскольку она может быть использована для хранения данных таких SCADA-систем, как InTouch и RSView32) используется встроенная библиотека System.Data.SqlClient.

Прототип системы

Было разработано программное обеспечение, которое реализует использование системы нечетких множеств и правил к ним. Были созданы классы для клиентов к ПЛК Siemens, Rockwell и базе данных Microsoft SQL Server, которые хранят в себе необходимую информацию для подключения к источникам данных и методы для получения информации. Созданы классы для хранения сведений о датчиках и регуляторах, значения которых используются для работы системы нечетких правил. Создан интерфейс для отображения значений выходных параметров системы при помощи элемента DataGridView, он показан на рисунке 3.

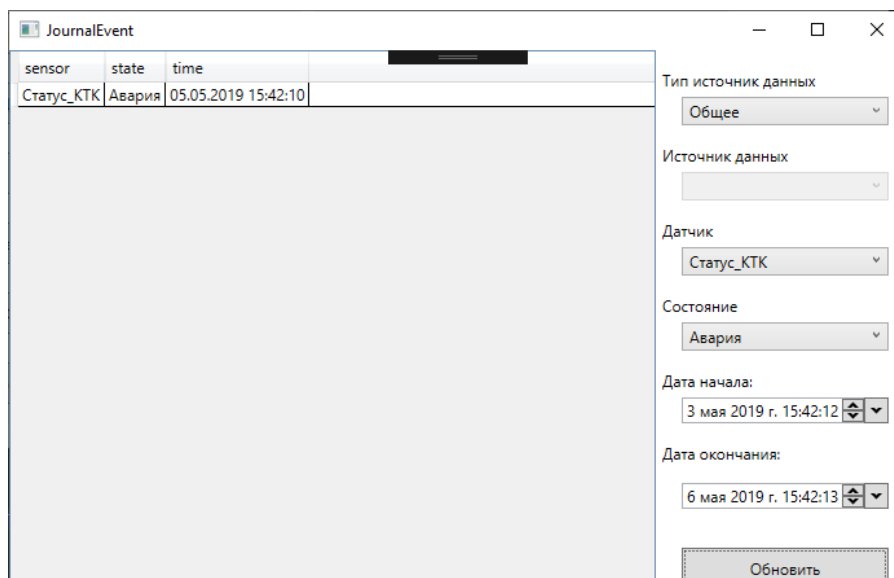


Рис. 3. Окно зафиксированных состояний выходных переменных

Создан интерфейс для определения того, состояния каких переменных должны сохраняться в локальной базе данных. В качестве локальной базы данных для системы используется SQLite.

Интерфейс создания правил для системы показан на рисунке 4.

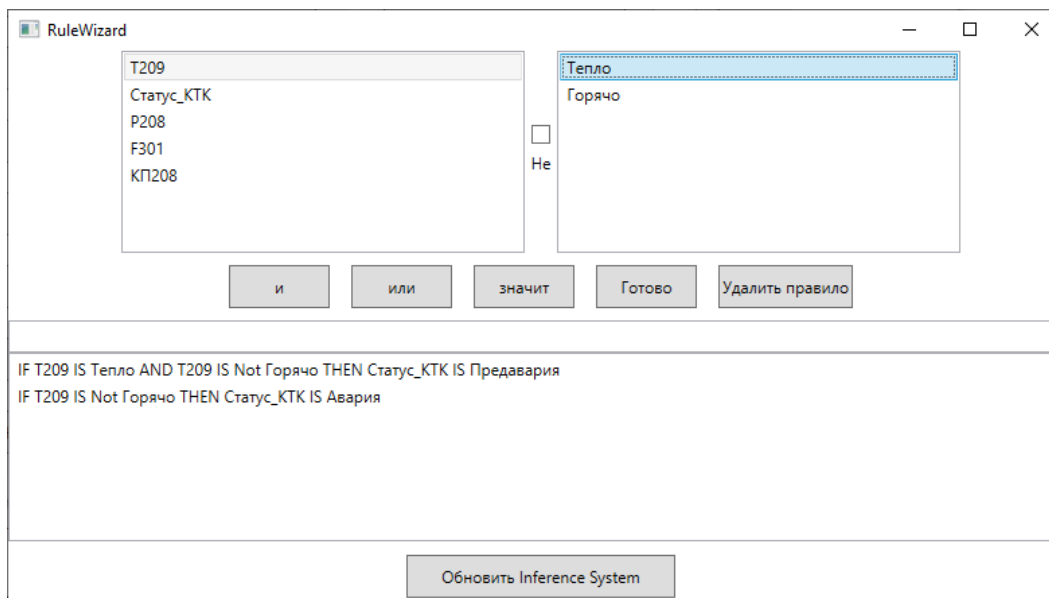


Рис. 4. Интерфейс задания правил

Для запуска системы в меню реализована кнопка, которая через задаваемый пользователем интервал асинхронно выполняет один цикл работы системы, показанный на рисунке 2.

Имеющийся на данный момент прототип реализует весь функционал, который описан в концепции, однако программа требует отладки в условиях работы с реальными данными, т.е. рекомендуется использовать хоть бы одно подключение к одному из ПЛК, для которых имеется класс клиента, для получения данных.

Выводы

Таким образом, разработан программный прототип системы, в которой применена теория нечетких множеств для получения аналитических данных по параметрам технологических узлов. В будущем после необходимых доработок она может использоваться как базовый инструментарий для разработки программного комплекса верхнего уровня АСУТП, адаптированный для технических средств, имеющихся на металлургическом производстве и не только.

Литература

1. Тагиев, К.Ф. Технологические инновации на машиностроительном предприятии: риски и пути их снижения / К.Ф. Тагиев // Вестник экономики, права и социологии, 2010. – №4. – С.64–69.
2. Трахтенгерц, Э.А. Компьютерные системы поддержки принятия управленческих решений / Э.А. Трахтенгерц // Проблемы управления, 2003. – №1. – С.13–28.
3. Втюрин, В.А. Автоматизированные системы управления технологическими процессами. Основы АСУТП / В.А. Втюрин – СПб. : Санкт-Петербургская Государственная Лесотехническая Академия имени С.М. Кирова, 2006. – 152 с.
4. Троелсен, Э. С# и платформа .NET / Э. Троелсен; Э. Троелсен. – СПб. : Питер, 2006. – 796 с.

УДК 004.021

КРАТКИЙ ОБЗОР ПЛАТФОРМЫ ARDUINO UNO И УЛЬТРАЗВУКОВОГО ДАТЧИКА ИЗМЕРЕНИЯ РАССТОЯНИЯ HC-SR04

Егурнов Н.О., Миненко А.С.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: ygurnovn@gmail.com

Аннотация:

Егурнов Н.О., Миненко А.С. Краткий обзор платформы arduino uno и ультразвукового датчика измерения расстояния hc-sr04. В данной статье, рассмотрены аппаратные возможности Arduino UNO и совместимого ультразвукового датчика измерения расстояний HC-SR04. Данные аппаратные решения будут использоваться, как основа для создания мобильного робота. Приведены технические характеристики аппаратной платформы. Определены перспективы применения в робототехнике.

Ключевые слова: ARDUINO, HC-SR04, робототехника, программное обеспечение, функциональное описание, навигация, ультразвуковой дальномер, управление движением.

Annotation:

Egurnov N.O., Minenko A.S. Overview platform arduino uno and ultrasonic distance measurement sensor hc-sr04. This article describes the hardware capabilities of the Arduino UNO and the compatible ultrasonic sensor for measuring distances HC-SR04. These hardware solutions will be used as the basis for creating a mobile robot. The technical characteristics of the hardware platform. Identified prospects for use in robotics.

Key words: ARDUINO, HC-SR04, robotics, software, functional description, navigation, ultrasonic range finder, motion control.

Общая постановка проблемы

В настоящее время в мире, возможно наблюдать процесс широкого применения робототехники. Ещё недавно, единственным видом роботов часто применяемых в жизни человека были промышленные машины, обусловленные дороговизной производства и сложностью в эксплуатации. Но, с течением времени, распространение роботов прикладного назначения стало доступно для персонального использования. Это обусловлено многими объективными факторами, решающими из которых, конечно же, является бурное развитие робототехники, информационных технологий и развитию элементов питания, способных достаточно долгое время поддерживать автономную работу устройства.

Сферы применения роботов и робототехники различны: образование, медицина, строительство, геодезия, метеорология и т.д. [1]. Из множества платформ для аппаратной реализации роботов, легких в освоении и не требующих больших финансовых затрат, выгодно использовать ту, которая обедняет в себе простоту и доступность. Платформу, имеющую огромное число комплектов дополнительных модулей, наборов датчиков и всех необходимых элементов включая документацию.

Постановка задачи

Для решения задач магистерской диссертации, требуется определиться с аппаратным обеспечением для построения мобильного робота, который в свою очередь является интеллектуальным агентом системы локализации и управления. Данная статья посвящена обзору платформы ARDUINO и ультразвукового дальномера HC-SR04. В ходе обзора требуется выявить технические характеристики аппаратных решений, удостоверится в их

пригодности для создания мобильного робота. Главными задачами, которых, обеспечивать стабильную работу сбор и передачу данных, на основе которых будет производиться локализация мобильного робота в помещении.

Функциональное описание ARDUINO UNO

Arduino – это удобная платформа для быстрой разработки электронных устройств, назначение которой упростить работу новичкам и предоставить широкий функционал профессионалам. Платформа пользуется популярностью благодаря удобству и простоте языка программирования. Язык программирования устройств Arduino основан на C/C++. Он прост в освоении, и на данный момент Arduino – это, возможно самый удобный способ программирования для устройств на микроконтроллерах, также благодаря открытой архитектуре и программному коду. Устройство программируется через USB без использования программаторов, что значительно облегчает и удешевляет его эксплуатацию и гибкость в применении.

Устройства на базе Arduino могут получать информацию об окружающей среде посредством различных датчиков, а также могут управлять различными внешними исполнительными устройствами.

Микроконтроллер на плате программируется при помощи языка Arduino, который был создан на основе языке Wiring. Среда разработки Arduino является развитием среды Processing, для функционального использования и простоты работы. Проекты устройств, основанные на Arduino, могут работать самостоятельно, либо же взаимодействовать с программным обеспечением на компьютере. Платы могут быть собраны пользователем самостоятельно или куплены в сборе. Исходные чертежи схем являются общедоступными, пользователи могут применять их по своему усмотрению [2].

Платы строятся на основе микроконтроллеров фирмы Atmel. В случае модели Arduino Uno это микроконтроллер ATmega328P. На платах присутствует линейный стабилизатор напряжения +5 В или +3,3 В. Тактирование осуществляется на частоте 16 МГц кварцевым резонатором. В микроконтроллере вшит загрузчик, следовательно, он не требует внешний программатор.

Платформа имеет 14 цифровых входов выходов, 6 из которых могут быть использованы как выходы ШИМ. Так же имеются 6 аналоговых входов, разъем USB, разъем питания, разъем ICSP и кнопку перезагрузки. Имеет 32 Кб Флеш-памяти, 0,5 Кб используется для загрузчика, 2Кб ОЗУ, 1 Кб EEPROM [3]. Имея достаточные вычислительные мощности и возможность подключения множества модулей к основной плате Arduino UNO, данное аппаратное решение является отличным ядром для реализации робота с широкими возможностями по применению.

Функциональное описание ультразвукового дальномера HC-SR04

Чаще всего бытовые роботы вовсе не оснащаются, какими либо датчиками или интеллектуальными системами обработки информации для навигации. Основанные на хаотичном перемещении устройства по неизвестной территории, используют соприкосновение робота с препятствиями как способ навигации. К примеру, для робота пылесоса не требуется точное картографирование рабочего пространства, вполне достаточно простого алгоритма действий. После физического контакта с препятствием робот попросту сменяет траекторию движения.

Для организации автономного движения мобильного робота с более сложной навигацией предлагается принцип, основанный на анализе результатов измерений ультразвуковым датчиком расстояний от робота до впереди стоящего препятствия, как в статическом положении, так и в процессе движения робота. Наличие препятствия в направлении движения робота определяется по времени прихода отраженного радиоимпульса, излученного и принятого датчиком в ультразвуковом диапазоне частот. Расчет дальности осуществляется по формуле:

$$x = \frac{ct}{2}, \quad (1)$$

где t – время прихода отраженного радиоимпульса с момента излучения;
 c – скорость распространения ультразвуковой волны в воздушном пространстве равное 340 м/с.

Ультразвуковой дальномер HC-SR04 является отличным техническим средством для реализации измерения расстояния. Основными характеристиками ультразвукового датчика являются, рабочее напряжение в 5.5 В. Потребляемый ток составляет 8 мА. Частота работы 40 кГц. Максимальная дистанция измерения расстояния 4 м при угле не более 15°. Размеры датчика: 37x20x15 мм [4]. В сочетании с дешевой данной устройством и его достаточным функционалом его использования в робототехнике является выгодным.

Выводы

В ходе данной статьи, посвященной обзору платформы ARDUINO и ультразвукового дальномера HC-SR04, выявлены технические характеристики, необходимые для реализации мобильного робота. Аппаратная платформа способна обеспечить стабильную работу по сбору и передаче необходимых данных, требующихся для локализации мобильного робота в помещении.

Подводя итог, по техническому оснащению платформа Arduino и ультразвуковой дальномер HC-SR04 идеально подходят для проектирования различных систем, в том числе и роботов благодаря понятной среде программирования, технической простоте и относительно не высокой цене. В дальнейшем данные аппаратные решения будут использоваться как основные модули для реализации аппаратной части магистерской диссертации, как демонстрационная модель реализуемых алгоритмов навигации на собранном мобильном роботе.

Стоит отметить, что перспективы развития мобильной робототехники можно представить в интеграции облачных сервисов и использовании искусственного интеллекта на базе самообучающихся нейронных сетей. Развитие, которых может обеспечить даже при ограниченной функциональности, размере и дешевизне роботов достаточно высокий практический потенциал, как группы интеллектуальных агентов, так и единичного робота. Создание более емких и компактных элементов питания повысит продолжительность автономной работы каждого из роботов, что положительно повлияет на их практическое использование в повседневной жизни и других прикладах областях.

Литература

1. Ивановский А.В. Начала робототехники: материал технической информации для подготовки методических пособий [Текст] / А.В. Ивановский // Минск: Вышей. шк., 1988. - 219 с.
2. Arduino.ru: Официальный сайт Arduino в России [Электронный ресурс] / Интернет-ресурс. - Режим доступа: <http://arduino.ru/>
3. Atmel.com: официальный сайт atmel [Электронный ресурс] / Интернет-ресурс. - Режим доступа: <http://www.atmel.com/Images/doc8161.pdf>
4. Конюх, В.Л. Основы робототехники: учеб. пособие для вузов по направлениям подготовки 220300 "Автоматизация технол. процессов и пр-в" и 220400 "Мехатроника и робототехника" [Текст] / В.Л Конюх // Феникс, 2008. - 282 с.

УДК 004.424

ИССЛЕДОВАНИЕ И РАЗРАБОТКА СИСТЕМЫ ДИАГНОСТИКИ ТЕХНИЧЕСКИХ ОБЪЕКТОВ

Орлов Ю.К., Свиридов Д.В.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: dennedswirriws@mail.ru

Аннотация:

Орлов Ю.К., Свиридов Д.В. Исследование и разработка системы диагностики технических объектов. Определена проблематика задачи. Рассмотрены основные понятия баз знаний и логического вывода. Определен алгоритм реализации обхода базы знаний, структура базы данных. Разработана блок-схема алгоритма работы программы.

Annotation:

Orlov Yu.K., Sviridov D.V. Research and development of a diagnostic system for technical objects. The problem is defined. The basic concepts of knowledge bases and logical inference are considered. The algorithm for implementing the crawling of the knowledge base and the database structure are defined. A flowchart of the program has been developed.

Постановка проблемы

Программное обеспечение для диагностирования необходимо в том случае, если система перестает нормально работать или если добавляется новое устройство. Даже при выполнении простых операций (например, смена оперативной памяти) или ищите неисправность, приведшую к сбою или «зависанию» системы, вам необходимо знать о компьютере больше, чем написано в прилагаемой к нему инструкции. Диагностические программы позволяют проверить работу как всей системы, так и отдельных ее узлов. Естественно, при эксплуатации системы необходимо регулярное техническое обслуживание. Именно это и служит залогом нормальной работы компьютера.

Цель работы – проектирование и разработка экспертной системы диагностики персонального компьютера, используя логический вывод по продукционной модели.

Реализация

В наши дни экспертные системы повсеместно используются для распространения опыта и знаний практически во всех сферах нашей жизни. Знания представляются в двух видах: личный и коллективный опыт. Если в предметной области знания представлены в виде коллективного опыта, то не имеет смысла использовать экспертную систему в данной области. Если же основная часть знаний является личным опытом экспертом данной области, при слабой структурированности знаний, данная область прекрасно подходит для использования экспертной системы.

Базы знаний тесно связаны с системами поиска и извлечения информации. Для этого необходима модель классификации и формат представления знаний. Наилучшим представлением понятий в базе знаний является иерархический способ или антология.

Онтологию некоторой области знаний вместе со сведениями о свойствах конкретных объектов часто называют «базой знаний». Базы знаний (в отличие от обычной базы данных) содержат в себе не только фактическую информацию, но и правила вывода, позволяющие

делать автоматические умозаключения об уже имеющихся или вновь вводимых фактах и тем самым производить семантическую (осмысленную) обработку информации.

В экспертных системах, которые используют прямой вывод по фактам, определяются заключения, что следуют из этих фактов. Если подобное заключение найдено, оно заносится в память. Метод прямого вывода зачастую применяется в системах диагностики. Данные выводы называются управляемые данными выводы. Логический вывод – это рассуждение, в ходе которого осуществляется переход от исходного суждения (высказывания или системы высказываний) с помощью логических правил к заключению – новому суждению (высказыванию или системе высказываний).

К логическому выводу обычно предъявляются (совместно или по отдельности) два основных требования:

- правила перехода должны воспроизводить отношение логического следования (ту или иную его разновидность);
- переходы в логическом выводе должны осуществляться на основе учёта только синтаксических характеристик высказываний или систем высказываний.

Был проведен ручной просчет методом обратного логического вывода. Целью ручной просчеты является наглядный результат работы метода для дальнейшей реализации в программном модуле.

Обратный порядок вывода – от заключений к фактам. В системах с обратным выводом вначале выдвигается некоторая гипотеза о конечном суждении, а затем механизм вывода пытается найти в рабочей памяти факты, которые могли бы подтвердить или опровергнуть выдвинутую гипотезу. Процесс отыскания необходимых фактов может включать достаточно большое число шагов, при этом возможно выдвижение новых гипотез (целей). Обратные выводы управляются целями. На рисунке 1 представлена схема прямого логического вывода.

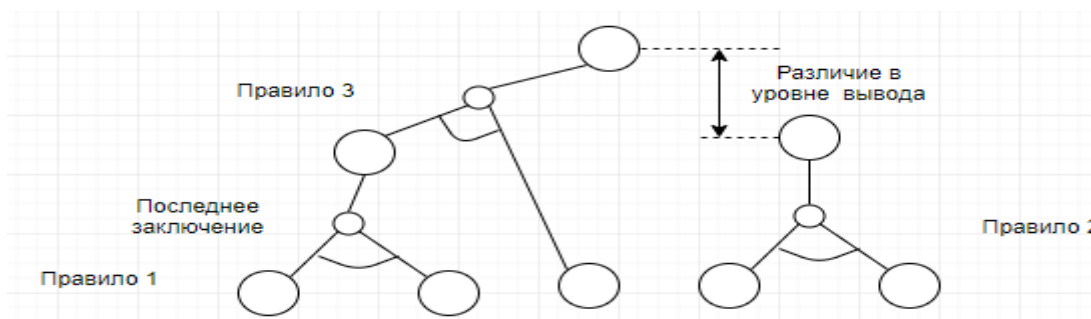


Рисунок 1 – Схема прямого логического вывода

Разрабатываемая программа будет состоять из трех модулей: модуль опроса пользователя, модуль обработки собранной информации и модуль обращения к базе данных. База данных будет представлена двумя таблицами. На рисунке 2 представлена схема базы данных.

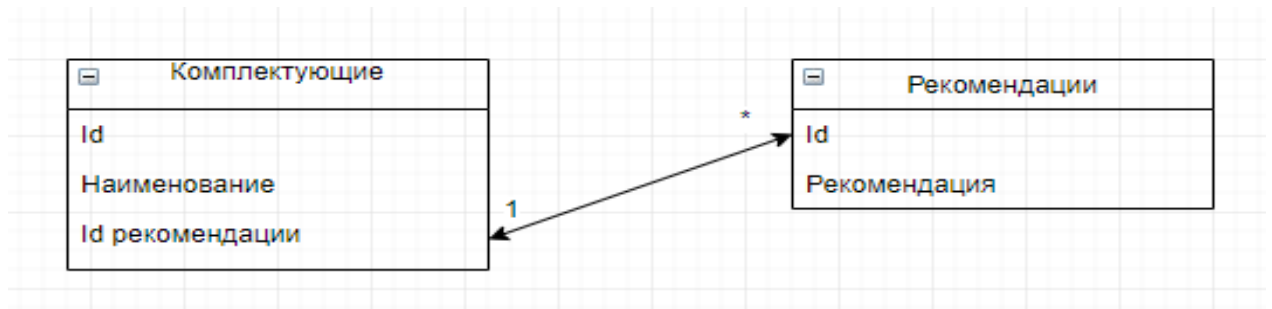


Рисунок 2 – Схема базы данных

На рисунке 3 представлена блок-схема алгоритма работы разрабатываемой программы.

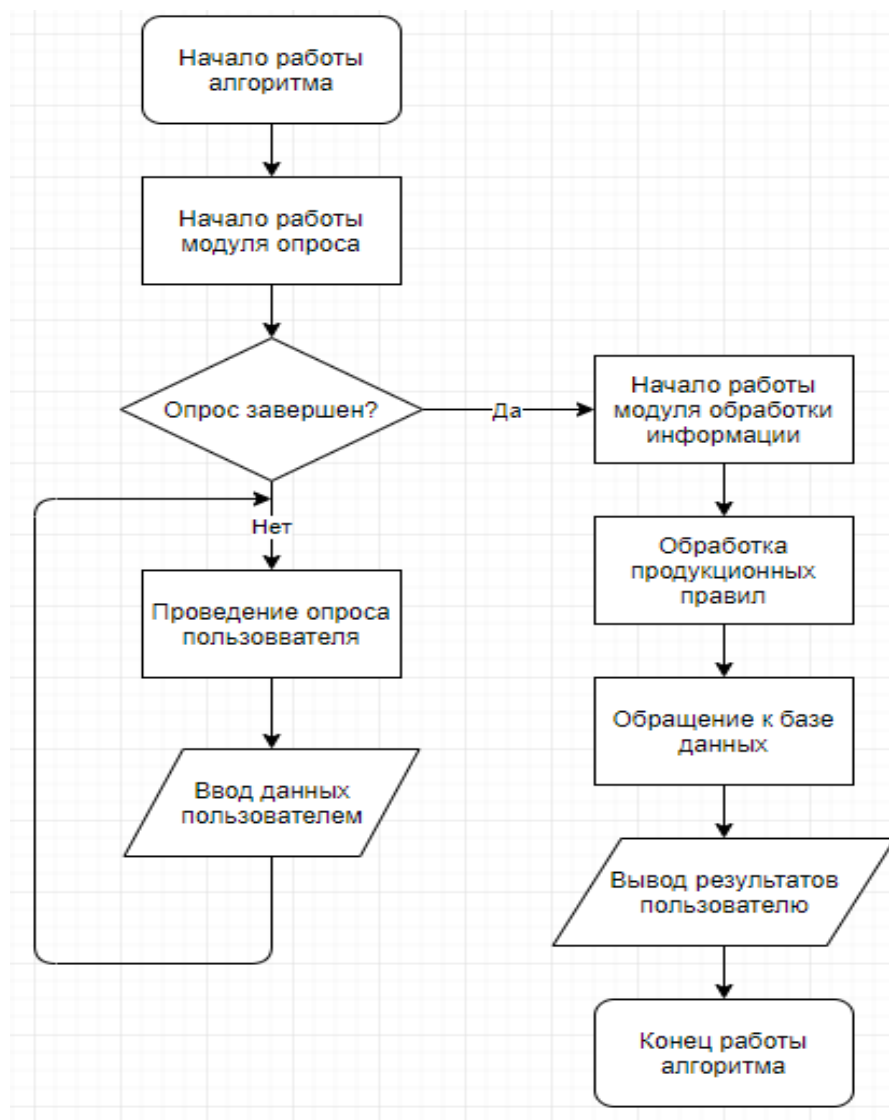


Рисунок 3 – Блок-схема алгоритма работы программы

Выводы

Проведенные исследования позволили выяснить, что для реализации системы необходимо использовать базу знаний. Для данной базы знаний необходимо разработать продукционную модель знаний, подкрепленную правилами вывода. Установлено, что в качестве метода логического вывода будет выбран прямой логический вывод.

Литература

1. Базы знаний – Википедия [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/База_знаний
2. Прямой и обратный логический вывод – Искусственный интеллект [Электронный ресурс]. – Режим доступа: <http://www.rriai.org.ru/pryamoy-i-obratnyiy-logicheskiy-vyivod-5.html>
3. Рихтер, Д. CLR via C#. Программирование на платформе Microsoft .NET Framework 2.0 на языке C# / Д. Рихтер, – СПб. : Питер, 2007. – 656с.
4. Нейгел, К. C# 2005 для профессионалов / К. Нейгел – К. : Вильямс, 2006. – 763 с.
5. Сравнительная характеристика, назначение и возможности современных языков – Студопедия [Электронный ресурс]. – Режим доступа: http://studopedia.ru/18_43813_sravnitel'naya-harakteristika-naznachenie-i-vozmozhnosti-sovremennih-yazikov.html
6. Грофф Р.Д. SQL: полное руководство / Р.Д. Грофф, Н.П. Вайнберг, Д.Э. Оппель – СПб. : Вильямс, 2014. – 961 с.
7. Труб, И. Объектно – ориентированное моделирование на C++: Учебный курс/ И. И. Труб. – СПб.: Питер, 2006. – 411 с.
8. Объектно-ориентированный анализ и проектирование с примерами приложений / [Гради Буч и др]. [3-е изд.] – М.: Вильямс, 2008. – 720с.

УДК 004.056.53 : 343.711.64

ЭЛЕКТРОННЫЙ ДОКУМЕНТООБОРОТ

Корнеева Е.А., Папин О.В., Едемская Е.Н., Сорокин Р.А.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: funtik-87@bk.ru

Аннотация:

Корнеева Е.А., Папин О.В., Едемская Е.Н., Сорокин Р.А. Электронный документооборот. Данная статья посвящена проблемам внедрения системы электронного документооборота в организациях.

Annotation:

Korneeva E.A., Papin O.V., Edemskaya E.N., Sorokin R.A. Electronic document management. This article is devoted to the problems of implementing the electronic document management system in organizations.

Общая постановка проблемы

В настоящее время, в период развития инновационной информационно-ориентированной экономики актуальным становится вопрос внедрения инновационных технологий в процесс управления, движения документов в организациях. Решение проблемы внедрения электронного документооборота в государственный аппарат управления взяло на себя Правительство.

Так, электронным документом, передаваемым по каналам связи, признается информация, подготовленная, отправленная, полученная или хранимая с помощью электронных, магнитных, оптических либо аналогичных средств, включая обмен информацией в электронной форме и по электронной почте.

Актуальным является внедрение систем электронного документооборота в организации, занимающиеся хозяйственной деятельностью.

Электронный документооборот

Электронный документооборот – единый механизм по работе с документами, представленными в электронном виде, реализацией концепции «безбумажного делопроизводства».

Электронная подпись (ЭП) – аналог собственноручной подписи, являющийся средством защиты информации, обеспечивающим возможность контроля целостности и подтверждения подлинности электронных документов.

Основные принципы электронного документооборота [1–3]:

- однократная регистрация документа, позволяющая однозначно идентифицировать документ;
- возможность параллельного выполнения операций, позволяющая сократить время движения документов и повышения оперативности их исполнения;
- непрерывность движения документа, позволяющая идентифицировать ответственного за исполнение документа (задачи) в каждый момент времени жизни документа (процесса);
- единая (или согласованная распределенная) база документной информации, позволяющая исключить возможность дублирования документов;

– эффективно организованная система поиска документа, позволяющая находить документ, обладая минимальной информацией о нем;

– развитая система отчетности по различным статусам и атрибутам документов, позволяющая контролировать движение документов по процессам документооборота и принимать управленческие решения, основываясь на данных из отчетов.

Электронный документооборот имеет свои достоинства и недостатки.

Внедрение систем электронного документооборота позволяет выполнить следующие задачи [4–6].

1. Полностью автоматизировать процесс работы с документами.
2. Обеспечить организацию работы с удаленными пользователями и группами пользователей.

3. Обеспечить интеграцию с внешними системами электронной почты.

4. Нарастивать базовые возможности системы электронного документооборота с помощью модульности и наличия встроенных инструментальных средств.

5. Обеспечить процесс одновременной работы в системе неограниченного числа пользователей.

6. Повысить эффективность работы с документами.

7. Повысить корпоративную культуру работы с документами.

8. Значительно уменьшить объемы бумажного документооборота.

9. Сократить время на обработку и пересылку документов, время поиска документов, время согласования и утверждения проектов документов.

10. Обеспечить все документационные процессы с одновременным использованием электронных и бумажных версий документов.

11. Упорядочить процесс регистрации всех видов документов (регистрация из электронной почты и web-форм, поддержка потокового сканирования, регистрация файлов любого формата).

12. Обеспечить управление потоками работ (передача документов между исполнителями).

13. Обеспечить процесс работы с взаимосвязанными документами.

14. Обеспечить управление документом на протяжении всего жизненного цикла.

15. Обеспечить прозрачность всех процессов с момента создания документа до момента его сдачи в архив.

16. Сохранение истории работы с документами (учет времени и авторов всех действий с документом, сохранение рабочих комментариев, поддержка версионности присоединенных файлов).

17. Проследить историю создания документа.

18. Обеспечить четкий процесс согласования и утверждения документов.

19. Выстроить отлаженную систему поручений.

20. Обеспечить четкий контроль исполнительской дисциплины.

21. Производить сортировку документов по любым критериям.

22. Обеспечить поиск информации по различным атрибутам и полнотекстовый поиск.

23. Обеспечить автоматизацию сбора и анализа статистических данных о движении документов.

24. Обеспечить архивное хранение электронных образов документов.

25. Обеспечить регламентацию прав доступа.

26. Обеспечить мгновенный доступ к документам.

27. Уменьшить количественные потери документов.

Так же система электронного документооборота имеет свои недостатки. Их стоит брать во внимание при принятии решений о внедрении систем электронного

документооборота. Если компания внедряет систему электронного документооборота с самого своего основания, то "стрессов" у сотрудников оно не вызовет. А если решение принимает компания, у которой уже достаточно долгое время действует обычный документооборот, то нужно принимать во внимание то, что реформирование системы может вызвать ряд "стрессов" у сотрудников. Все новое, принимается с трудностями.

Также, нужно принять во внимание тот факт, что компания может понести затраты на приобретение программ и систем документооборота, а также на их внедрение и дальнейшее обслуживание.

Также существуют пробелы в информационной безопасности [7-9]. Так как рынок наводнен "пиратскими" дисками с перекачанными базами данных серьезных государственных структур, имеющих хорошую защиту, что можно говорить о предприятиях мелкого и среднего бизнеса. Конкуренция – закон бизнеса. Недобросовестному конкуренту получить информацию, размещенную на бумажных носителях, сложнее: документы могут храниться в разных помещениях, шкафах (столах, сейфах), разрозненных папках. С электронными базами данных проще. Современные хакеры посредством удаленного доступа взламывают дорогостоящие программы с высочайшей степенью защиты.

Построить полный эмулятор современного электронного ключа [10] – это достаточно трудоемкий процесс, требующий большого количества времени и существенных инвестиций. Ранее злоумышленникам это удавалось: например, компания Aladdin признает, что в 1999 году злоумышленникам удалось разработать довольно корректно работающий эмулятор ключа HASP3 и HASP4. Это стало возможным благодаря тому, что ключ использовал проприетарный алгоритм кодирования, который был взломан. Сейчас большинство ключей используют публичные криптоалгоритмы, поэтому злоумышленники предпочитают атаковать какой-то конкретный защищенный продукт, а не защитный механизм в общем виде. Для современных систем защиты HASP эмуляторов в свободном доступе нет, так как используется криптосистема с открытым ключом.

Существующие табличные эмуляторы реализованы только для конкретных приложений. Возможность их создания была обусловлена неиспользованием (или неграмотным использованием) основного функционала электронных ключей разработчиками защит.

Следующий недостаток – это резкое увеличение потока документооборота. Как результат – серверы не справляются, падает производительность труда, что требует дополнительных вложений во внедрение нового программного обеспечения, а именно, вложений в новое аппаратное (серверное) обеспечение. При наличии бумажного документооборота такой резкий рывок невозможен.

И еще один недостаток – это увеличение трудозатрат как следствие увеличения документооборота. Работодатель не успевает адекватно реагировать на подобные скачки в принятии решений по кадровым вопросам. Объемы возрастают, тогда как штатный состав остается прежним.

Основной проблемой внедрения системы электронного документооборота является модернизация технической инфраструктуры, в том числе приобретение необходимого оборудования (источники бесперебойного питания, серверы и т.д.) и программного обеспечения (ПО). Затем проводится установка программного обеспечения на сервер и рабочие места пользователей, а также настройка системы в соответствии с организационной структурой компании.

Другой проблемой является обучение персонала навыкам работы с системой электронного документооборота. Этап обучения часто совмещается с началом эксплуатации системы. Как правило, пользователи получают базовые знания по работе в системе электронного документооборота в группах, формируемых в зависимости от их роли в

системе. Полученные знания закрепляются на рабочих местах при консультационной поддержке специалистов, проводивших внедрение системы электронного документооборота.

Опытная эксплуатация системы документооборота проводится с целью выявления скрытых отклонений от технического задания и неучтенных требований. В случае их выявления проводится соответствующая корректировка настроек системы электронного документооборота.

Выводы

Процесс внедрения системы электронного документооборота, независимо от сферы деятельности организации, численности ее персонала и территории присутствия – сложный многоэтапный процесс, который необходимо реализовывать во всех организациях, так как данная система позволит систематизировать, облегчить и повысить эффективность работы всей организации.

Литература

1. Глекова, Н.Л., Жильников А.Ю. Анализ процесса реализации политики импортозамещения // В сборнике: Инновационные подходы к решению социально-экономических, правовых и педагогических проблем в условиях развития современного общества материалы международной научно-практической конференции. 2015. С. 139–142.
2. Науменко, Т.В. Маркетинг и маркетинговые коммуникации в системе современной экономики // Конкурентоспособность в глобальном мире: экономика, наука, технологии. 2016. № 3 (15). С. 171–176.
3. Жильников, А.Ю. Региональная инновационная система // В сборнике: Теоретические и прикладные вопросы науки и образования сборник научных трудов по материалам Международной научно-практической конференции: в 16 частях. 2015. С. 62–65.
4. Андреева, В.И. Делопроизводство. Требования к документообороту фирмы / В.И. Андреева. – М.: ИНФРА 1996. – 178 с.
5. Андреева, В.В. Как правильно подготовить и оформить приказ / В.В. Андреева. – М.: ИНФРА – М, 1997. – 94 с.
6. Герасименко В.А. Защита информации в автоматизированных системах обработки данных / В.А Герасименко. – М.: Энергоатомиздат, 1994. – 124 с.
7. ГОСТ 16487-83. Делопроизводство и архивное дело. Термины и определения. – М.: Издательство стандартов, 1983.
8. ГОСТР 6.30-97 Унифицированные системы документации. Унифицированная система организационно-распорядительной документации. Требования к оформлению документов. – М.: Издательство стандартов, 1997.
9. Государственная система документационного обеспечения управления. Основные положения, Общие требования к документам и службам документационного обеспечения. – М.: ВНИИДАД, 1991.
10. Бочаров, Д.М. Компьютерные преступления эстафетным методом [Текст] / Д.М. Бочаров, Р.А. Сорокин, Е.А. Норкене // Цифровизация экономики и общества: проблемы, перспективы, безопасность: Материалы международной научно-практической конференции 29 марта 2019 года. – В 2-х т. - Т.2: – ГОУ ВПО «Донбасская юридическая академия» / Отв. ред.: к.э.н., доц. И.П. Подмаркова, Б.Б. Шавыркин. – Донецк, «Цифровая типография», 2019. – С. 28-32.

УДК 004.896

СОЗДАНИЕ ПРОГРАММЫ ДЛЯ РОБОТА И РЕАЛИЗАЦИЯ ЕГО ФУНКЦИОНИРОВАНИЯ

Перепелица Р.А., Давиденко А., Куликов Д.Р., Елисеев В.И.
Донецкий национальный технический университет, г. Донецк
кафедра искусственного интеллекта и системного анализа
E-mail: perepelitsa.r.a@gmail.com

Аннотация

Перепелица Р.А., Давиденко А.Н., Куликов Д.Р., Елисеев В.И. Создание программы для робота и реализация его функционирования. Проанализированы существующие модели роботов и возможные подходы к решению задачи по участию в соревновании роботов «Кегельринг». Построена конструкция робота, пригодная для решения поставленной задачи. Определена среда, в которой будет удобнее разрабатывать, читать и сопровождать программу для робота. Разработан алгоритм и написана программа в среде NXT Programming для выбивания всех кегель, находящихся на ринге.

Ключевые слова: робот, программирование, кегельринг.

Annotation

Perepelitsa R.A., Davidenko A.N., Kulikov D.R., Eliseev V.I. Creating a program for the robot and the implementation of its operation. The existing models of robots and possible approaches to solving the problem of participation in the Kegeling robots competition are analyzed. Built model of the robot, suitable for solving the problem. Defined environment in which it will be more convenient to develop, read and accompany the program for the robot. The algorithm was developed and the program was written in the NXT Programming environment for knocking out all the kegel in the ring.

Keywords: robot, programming, kegelring.

Постановка проблемы. В мире робототехники проводится множество различных соревнований. Наиболее известными и зрелищными являются сумо роботов, поиск выхода из лабиринта, скоростное движение по линии и танцы роботов. Большое количество соревнований проводятся на всемирной олимпиаде роботов (WRO), где перед роботами ставятся сложные интеллектуальные задачи. Мы рассмотрим одну из соревновательных задач начального уровня - так называемый "Кегельринг". Для участия в соревнованиях Кегельринг по робототехнике необходимо разработать робота, который может быть собран из LegoNXT. В связи с этим существует ряд проблем, которые требуют решения. Прежде всего, необходимо построить поле для кегельринга из подручных средств. Затем нужно разработать робота, модель которого должна соответствовать поставленной задаче. После этого необходимо выбрать среду программирования, разработать алгоритм и отладить программу управления.

Цель статьи – ознакомить с процессом создания робота для участия в соревнованиях по кегельрингу.

Смысл соревнования. Смысл соревнования состоит в том, что робот за наиболее короткое время, не выходя за пределы круга, очерчивающего ринг, должен вытолкнуть расположенные на ринге кегли. Диаметр ринга - 1 метр. Цвет ринга - белый, цвет ограничительной линии - черный. Ширина ограничительной линии - 5 см. Ринг может быть

сделан из листов белого ватмана, а ограничительная линия нарисована тушью. Пример ринга изображен на рисунке 1. В качестве кеглей можно использовать пустые жестяные банки объемом 0,5 л. из-под напитков. Перед началом соревнования на ринге устанавливается 8 кеглей. Робот помещается в центр ринга и включается, после этого в его работу нельзя вмешиваться. Максимальная ширина робота не должна превышать 20 см, длина тоже не должна быть больше 20 см. Высота робота не ограничена. Робот должен выталкивать кегли своим корпусом, применение механических или пневматических устройств для выталкивания кеглей запрещено. Кегля считается вытолкнутой, если никакая ее часть не находится внутри круга, ограниченного линией.

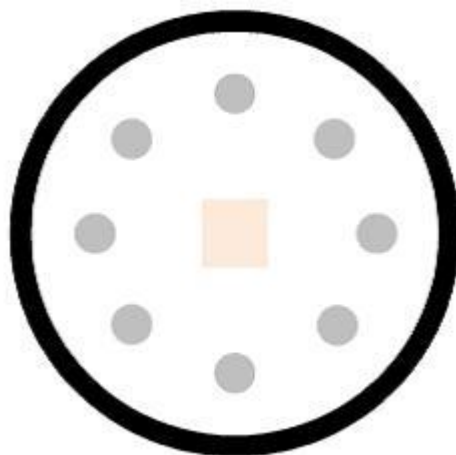


Рис. 1. Поле для соревнования "Кегельринг"

Алгоритм решения задачи.

1. Робот, находясь в центре поля, начинает вращаться по часовой стрелке, пока не заметит кеглю.

2. Двигаясь в направлении кегли, робот выталкивает её за пределы окружности.

3. Заметив черную границу поля, робот возвращается назад, в место старта.

Следовательно, наш робот должен:

1. Уметь вращаться на месте вокруг своей оси.

2. Уметь двигаться прямолинейно.

3. Уметь обнаруживать предмет, удаленный на некоторое расстояние.

4. Уметь обнаруживать границу поля.

5. Уметь подсчитывать количество оборотов.

Необходимые компоненты. Данные требования диктуют нам условия конструкции робота:

1. Для реализации первых двух условий можно использовать робота, использующего два мотора, собранного по стандартной инструкции для Legomindstorm.

2. Для обнаружения кегли можно воспользоваться имеющимся в наличии ультразвуковым датчиком;

3. Границу поля поможет обнаружить датчик освещенности.

Программирование робота. Основным программным обеспечением для программирования робота LegoMindstormNXT является NXTprogramming. В данной среде разработки реализован визуальный способ проектирования программ, что очень удобно для работы. Программа составляется из блоков. Каждый блок представляет различные типы действий.

Для вращения вокруг своей оси используется следующий метод: первый двигатель вращается в одну сторону, второй – в обратную. Во время разворота вокруг своей оси считываются данные с ультразвукового датчика расстояния и, когда робот понимает, что впереди кегля, вращение заканчивается и он едет вперед. Ультразвуковой сенсор заменяет роботу зрение. Он работает по тому же принципу, что и локатор летучей мыши: измеряет расстояние путём расчёта времени, которое потребовалось звуковой волне для возвращения после отражения от объекта. Как следует из его настроек, датчик способен измерять расстояние от 0 до 255 см с точностью +/- 3 см. Однако на практике минимальное расстояние, на котором ультразвуковой датчик выдаёт осмысленные показания составляет около 5 - 7 см. Кроме того, следует иметь в виду, что по сравнению с датчиками цвета, света, касания и др., это медленный датчик. Во-первых, это объясняется малой скоростью звука по сравнению со скоростью света, а во-вторых, медленным протоколом обмена данными, используемым для этого датчика.

Если кегля найдена, то алгоритм её выталкивания состоит в следующем:

1. Двигаться вперёд до чёрной линии.
2. Вернуться назад.

Во время движения вперед происходит считывание данных с датчика освещенности. Чтобы лучше представить себе, что "видит" датчик освещенности, вообразите, что он перемещается над чёрно-белой фотографией. Результатом измерения будет яркость в диапазоне от 0 (чёрный цвет) до 100 (белый цвет), хотя крайних значений на практике зафиксировать не удастся. Если значение, полученное с этого датчика, будет мало, т. е. робот наехал на черную черту, то движение завершается и робот должен вернуться на исходную позицию. Возникает вопрос: насколько сильно робот должен возвращаться назад? Очевидно, на то же расстояние, которое он проехал до линии. Как определить это расстояние?

Моторы LegoMindstorms содержат встроенный датчик оборотов. Снимать показания этого датчика можно при помощи блока RotationSensor из меню Sensor. Этот блок подсчитывает количество градусов (с точностью до 1°) или количество полных оборотов вала мотора.

Выводы. Проанализированы существующие модели роботов и возможные подходы к решению задачи по участию в соревновании роботов «Кегельринг». Построена модель робота, пригодная для решения поставленной задачи. Определена среда, в которой будет удобнее разрабатывать, читать и сопровождать программу для робота. Разработан алгоритм и написана программа в среде NXTProgramming для выбивания всех кегель, находящихся на ринге.

Список литературы

1. Методические указания к лабораторным занятиям «Программирование LegoMindstormNXT» // Томский государственный университет систем управления и радиотехники: [Томск]. [2012].
2. Введение в программирование LEGO-роботов на языке NXT-G [электронный ресурс] // Intuit.ru [сайт] [2019] URL: <http://www.intuit.ru/studies/courses/14007/1280/info>
3. Мультибот: Базовое транспортное средство [электронный ресурс] // Prorobot.ru [сайт] [2019] URL: http://www.prorobot.ru/lego/multibot_transport.php
4. Mindstorms (серия LEGO) [электронный ресурс] // Википедия – свободная энциклопедия [сайт] [2019] URL: [https://ru.wikipedia.org/wiki/Mindstorms_\(серия_LEGO\)](https://ru.wikipedia.org/wiki/Mindstorms_(серия_LEGO))
5. LegoMindstorms Руководство пользователя [электронный ресурс] // Mindstorms.ru [сайт] [2019] URL: http://www.mindstorms.ru/img/file/8547_Mindstorms.pdf

УДК 004.896

СОЗДАНИЕ ПРОГРАММЫ ДЛЯ УЧАСТИЯ В СОРЕВНОВАНИИ «КЕГЕЛЬРИНГ»

Перепелица Р.А., Давиденко А. Куликов Д.Р. Елисеев В.И.

Донецкий национальный технический университет, г. Донецк
кафедра искусственного интеллекта и системного анализа

E-mail: kulikovWww@gmail.com

Аннотация

Перепелица Р.А., Давиденко А.Н., Куликов Д.Р., Елисеев В.И. Создание программы для участия в соревновании «Кегельринг». Проанализированы существующие модели роботов и возможные подходы к решению задачи по участию в соревновании роботов «Кегельринг». Построена конструкция робота, пригодная для решения поставленной задачи. Определена среда, в которой будет удобнее разрабатывать, читать и сопровождать программу для удалённого управления роботом. Разработана программа для управления роботом с ПК в среде Visual Studio с помощью которой, пользователь управляя роботом, сам выбивает все кегели, находящиеся на ринге.

Ключевые слова: робот, программирование, кегельринг.

Annotation

Perepelitsa R.A., Davidenko A.N., Kulikov D.R., Eliseev V.I. Creating programs for robots and the implementation of its operation. The existing models of robots and possible approaches to solving problems of participation in the Kegeling robots competitions are analyzed. The construction of the robot, suitable for solving the task. A specific environment in which it will be convenient to develop, read and maintain a program for remote control of the robot. A program has been developed for controlling the robot from a PC in the Visual Studio environment, with the help of which the user controls the robot.

Keywords: robot, programming, kegelring

Постановка проблемы. В мире робототехники проводится множество различных соревнований. Наиболее известными и зрелищными являются сумо роботов, поиск выхода из лабиринта, скоростное движение по линии и танцы роботов. Большое количество соревнований проводятся на всемирной олимпиаде роботов (WRO), где перед роботами ставятся сложные интеллектуальные задачи. Мы рассмотрим одну из соревновательных задач начального уровня - так называемый "Кегельринг". Для участия в соревнованиях Кегельринг по робототехнике необходимо разработать робота, который может быть собран из Lego NXT. В связи с этим существует ряд проблем, которые требуют решения. Прежде всего, необходимо построить поле для кегельринга из подручных средств. Затем нужно разработать робота, модель которого должна соответствовать поставленной задаче. После этого необходимо выбрать среду программирования и разработать программу для удалённого управления роботом с персонального компьютера.

Цель статьи – ознакомить с процессом создания программы для удалённого управления роботом с персонального компьютера.

Смысл соревнования. Смысл соревнования состоит в том, что робот за наиболее короткое время, не выходя за пределы круга, очерчивающего ринг, должен вытолкнуть расположенные на ринге кегли. Диаметр ринга - 1 метр. Цвет ринга - белый, цвет ограничительной линии - черный. Ширина ограничительной линии - 5 см. Ринг может быть

сделан из листов белого ватмана, а ограничительная линия нарисована тушью. В роле кегель выступают банки из-под напитков объёмом 0,5 литра. Пользователь, управляя роботом, должен сталкивать их за пределы кегельринга и возвращать робота в центр, после чего, повторять эту процедуру до тех пор, пока в кегельринге не окажется не единой кегли.

Создание программы. Для разработки программы была выбрана среда программирования Visual Studio 2017, как наиболее удобная и подходящая под данный тип задач. И библиотеки, необходимые для подключения к NXT блоку и управления им. Для того, чтобы понять, когда нужно остановиться пользователю, необходимо следить за обновлением показателей датчика освещённости, когда его показатели уменьшаются до определённых единиц (34 и ниже), тогда необходимо возвращаться в центр кегельринга. Для того, чтобы понять на сколько оборотов нужно вернуться, необходимо следить за их количеством до черной линии, после чего необходимо обнулить количество оборотов нажатием соответствующей кнопки и двигаться далее до центра, продолжая эту процедуру до конца. Функция, задающая интервал частоты обновления датчиков показано на рисунке 1.

```
try
{
    brick = new Brick<Sensor, Sensor, Sensor, Sensor>("com3");
    brick.Connection.Open();
    // Подключение датчиков через порты и чтение данных с них
    brick.Vehicle.LeftPort = MotorPort.OutB;
    brick.Vehicle.RightPort = MotorPort.OutC;
    brick.Vehicle.ReverseLeft = false;
    brick.Vehicle.ReverseRight = false;

    brick.Sensor3 = new NXTLightSensor(LightMode.On);
    brick.Sensor4 = new Sonar(SonarMode.Centimeter);

    timer.Tick += new EventHandler(TimerEventProcessor);
    timer.Interval = 100; // Интервал обновления датчиков 100 миллисекунд
    timer.Start();
}
```

Рис. 1. Функция, задающая интервал частоты обновлений датчиков и чтение информации с них

Для того, чтобы пользователь мог управлять движением робота по нажатию клавиш с персонального компьютера, необходимо задать их в программе вместе со скоростью оборотов, с которой робот будет передвигаться по нажатию клавиш пользователем. Функция, задающая клавиши движения и скорость робота, изображена на рисунке 2.

```
sbyte speed = 50; // Скорость движение робота
switch (direction)
{
    case 'w': // Нажмите W для движение вперёд
        brick.Vehicle.Forward(speed);
        running = true;
        break;
    case 's': // нажмите S для движения назад
        brick.Vehicle.Backward(speed);
        running = true;
        break;
    case 'd': // Нажмите D для движения вправо
        brick.Vehicle.SpinRight(speed);
        running = true;
        break;
    case 'a': // Нажмите A для движения влево
        brick.Vehicle.SpinLeft(speed);
        running = true;
        break;
}
```

Рис. 2. Функция, задающая движение по клавишам и скорость робота

Для того, чтобы было удобно управлять роботом, необходимо оформить интерфейс управления. Самым удобным способом для этого, является создание интерфейса с помощью библиотеки Windows Forms. Для этого создадим пустую форму и наложим на неё все

необходимы элементы. Такие как button (кнопки), text box (поле ввода/вывода текста) и label (текстовые поля для обозначения), а также, сделаем инструкцию по управлению роботом справа от всех полей ввода. Сделаем это так, как показано на рисунке 3.

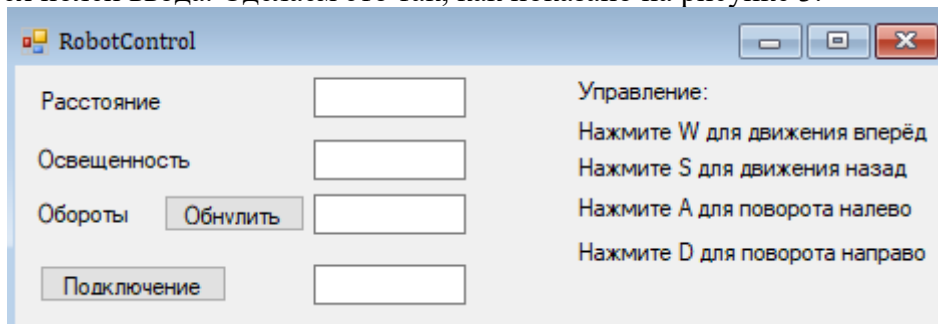


Рис. 3. Интерфейс программы

Выводы. Проанализированы существующие модели роботов и возможные подходы к решению задачи по участию в соревновании роботов «Кегельринг». Создано программное обеспечение, пригодное для управления пользователем с персонального компьютера. Определена среда, в которой будет удобнее разрабатывать программное обеспечение для управления с персонального компьютера роботом. Разработана и написана программа в среде Visual Studio с оформленным интерфейсом, для возможности выбивания всех кегель пользователем, использующим только свой персональный компьютер не смотря на кегельринг с роботом, а ориентируясь только по показателям датчиков на мониторе персонального компьютера.

Литература

1. Методические указания к лабораторным занятиям «Программирование LegoMindstormNXT» // Томский государственный университет систем управления и радиотехники: [Томск]. [2012].
2. Введение в программирование LEGO-роботов на языке C# [электронный ресурс]// Intuit.ru [сайт] [2018] URL: <http://www.intuit.ru/studies/courses/14008/1270/info>
3. Мультибот: Базовое транспортное средство [электронный ресурс]//Prorobot.ru [сайт] [2019] URL:http://www.prorobot.ru/lego/multibot_transport.php
4. Рихтер Дж. CLR via C#. Программирование на платформе Microsoft .NET Framework 2.0 на языке C#. Мастер-класс. /Дж. Рихтер. - Пер. с англ. — 2-е изд., 2008. — 656 с.
5. Седжвик Роберт. Фундаментальные алгоритмы на C / Роберт Седжвик

УДК 004.89

ПРОГНОЗ И ОЦЕНКА РИСКА ВОЗНИКНОВЕНИЯ ЧРЕЗВЫЧАЙНОЙ СИТУАЦИИ НА УГЛЕДОБЫВАЮЩИХ ПРЕДПРИЯТИЯХ

Г.А. Резникова, Ю.К. Орлов

Донецкий национальный технический университет, г. Донецк
кафедра искусственного интеллекта и системного анализа

E-mail: g_reznikova@list.ru

Аннотация

Резникова Г.А., Орлов Ю.К. Прогноз и оценка риска возникновения чрезвычайной ситуации на угледобывающих предприятиях. Сделан анализ организационных рисков всех уровней иерархической структуры предприятия с помощью формирования качественных информационных потоков. Разработан порядок проведения анализа опасностей и оценки риска аварий.

Annotation

Reznikova G.A., Orlov Yu.K. Forecast and risk assessment of an emergency situation. The analysis of organizational risks of all levels of the hierarchical structure of an enterprise has been made using the formation of high-quality information flows. A procedure for conducting hazard analysis and risk assessment of accidents has been developed.

Постановка задачи.

Одной из задач обеспечения безопасности производства является снижение риска возникновения аварий и катастроф на угольных шахтах от взрыва метановоздушной смеси. Известно, что от взрыва метана в шахтах погибает большое количество людей и наносится значительный моральный и материальный ущерб не только горным предприятиям, но и обществу в целом.

Современные угольные шахты – особо опасные производственные объекты, одним из основных источников опасности в которых является метан, выделяющийся в горные выработки в процессе выемки угля. Борьба с взрывами метана является актуальнейшей проблемой обеспечения безопасных условий труда шахтеров.

Цель статьи – сокращение числа возникновения чрезвычайных ситуаций, снижение производственного травматизма и обеспечение безопасности производственных процессов, оборудования, зданий и сооружений на угледобывающих предприятиях, за счет разработки системы контроля и мониторинга состояния промышленной безопасности, позволяющей спрогнозировать и дать оценку риску возникновения аварий, которая в свою очередь позволит минимизировать время на принятие управленческих решений для предотвращения катастрофы.

Анализ организационных рисков всех уровней иерархической структуры предприятия.

Высший уровень иерархии – руководство горного предприятия. В обязанности руководства входит управление деятельностью всей компании, формирование ее стратегии, определение целей, миссии, основных принципов взаимодействия сотрудников с клиентами, партнерами, подрядчиками и между собой [1].

Второй уровень иерархии – управление производством. На этом уровне от сотрудника требуются глубокие знания в конкретной области производственной деятельности [1].

На третьем уровне иерархии работнику достаточно тех знаний и умений, которые необходимы ему для выполнения своей функции [3].

Исходя из этого, функции управления, в зависимости от ступени иерархии, различаются:

- по масштабам и значимости последствий в случае принятия неверного управленческого решения;
- по объему информации, необходимой для принятия управленческого решения.

На любом иерархическом уровне управления необходимо достаточное для принятия решения количество своевременно получаемой информации заданной точности. Обеспечение управленцев информацией имеет большое значение для создания безопасных условий труда на горных предприятиях [2].

Наличие достоверных сведений о состоянии производственной системы, оцениваемом по состоянию нескольких смежных взаимосвязанных сред, дает возможность принимать решения о дальнейшем развитии горных работ, об объемах добычи, об оптимальных режимах работы техники. Общий поток информации распределяется по иерархическим уровням в соответствии с потребностями и полномочиями управленцев [4].

Работа над повышением качества информационных потоков должна проводиться в следующих направлениях:

- выявление и ликвидация причин сокрытия или искажения информации на всех уровнях иерархии;
- стимулирование предоставления достоверной информации;
- повышение квалификации работников;
- дублирование информационных потоков;
- создание точек входа и выхода информационных каналов в опасных зонах;
- минимизация сроков предоставления информации [5].

Снизить организационные риски угольного предприятия можно поэтапно несколькими способами:

- с помощью аттестационных мероприятий, организации обучения персонала;
- выстраивая систему мотивации персонала экономическими и психологическими методами;
- налаживая циркуляцию информационных потоков со сведениями высокой достоверности;
- с помощью автоматизации обработки и анализа информации о промышленной безопасности;
- разрабатывая внутрикорпоративные технологические стандарты компании;
- с помощью регулярного внутреннего аудита безопасности решений по изменению технологий, введения новой техники и т.д.;
- организуя научное сопровождение и независимую экспертизу всех проектов строительства, реконструкции, развития [5].

Принципами сформированного системного подхода к снижению организационных рисков угледобывающей компании на всех уровнях иерархии являются:

- использование как технико-технологических, так и организационных методов в борьбе с рисками;
- системное проектирование с целью снижения фоновых рисков;
- достаточность обеспечения средствами безопасности и другими ресурсами;
- достоверность информационных потоков;
- наблюдение за поведением горного массива в динамике и корректировка технических решений в соответствии с его реакцией на воздействия;
- повышение значимости безопасности производства, информирование и повышение квалификации персонала [5].

Порядок проведения анализа опасностей и оценки риска аварий.

Результаты анализа риска используются при декларировании промышленной безопасности опасных производственных объектов (далее – ОПО) [5].

Процесс проведения анализа включает следующие этапы:

- планирование и организация анализа риска аварий;
- идентификацию опасностей аварий;
- оценку риска аварий;
- определение степени опасностей и (или) наиболее опасных составных элементов;
- разработку рекомендаций по снижению аварий.

Общая схема проведения анализа опасностей и оценки риска аварий на ОПО представлена на рисунке 1.

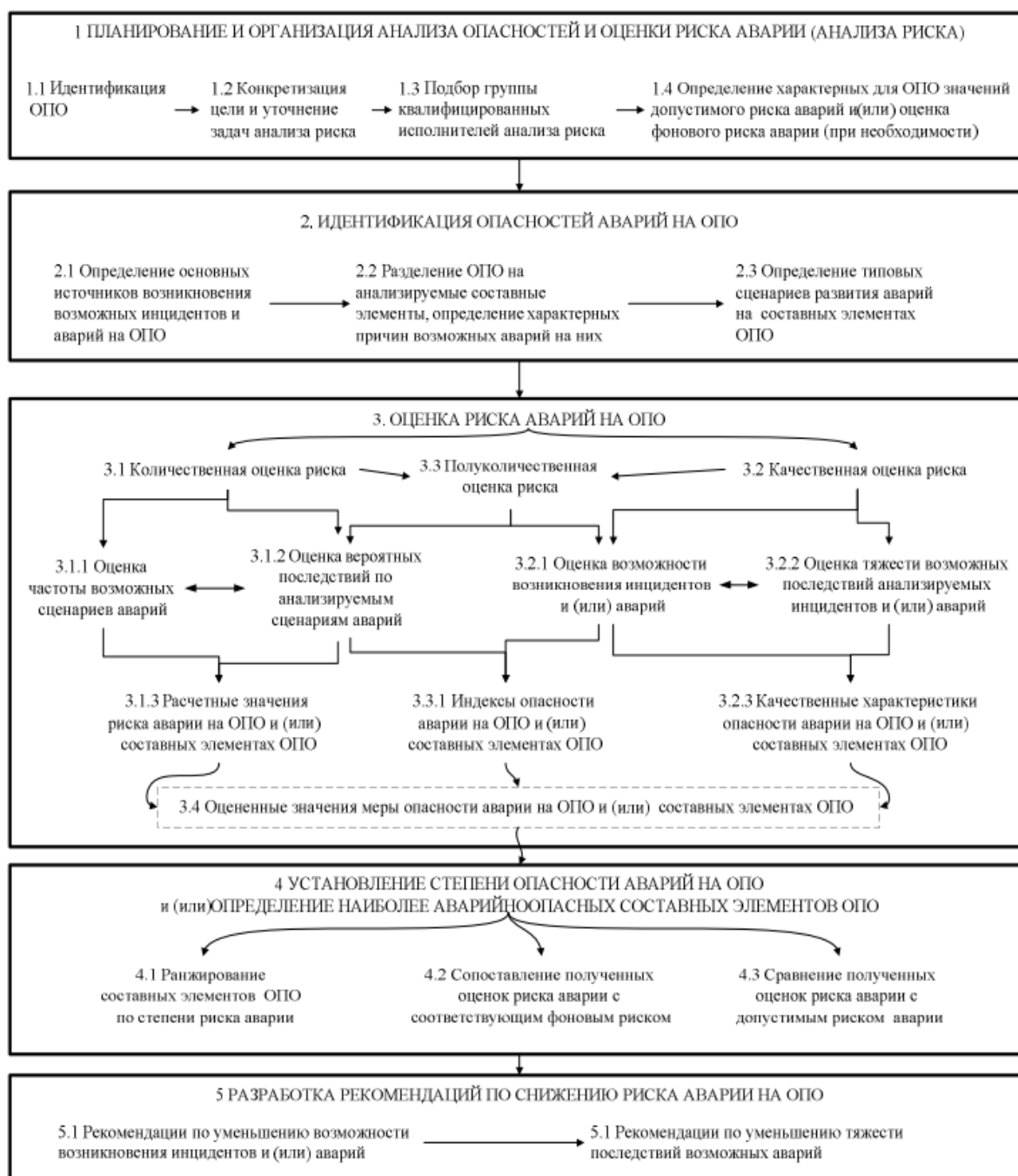


Рисунок 1 – Блок-схема проведения анализа опасностей и оценки риска аварий на ОПО

В исследованиях по проблеме риска возникло отдельное направление работ под общим названием «Управление риском». Особенностью этого направления является комплексность, включающая в себя различные аспекты-технические, организационно-управленческие, социально-экономические, медицинские, биологические и др.

С целью снижения риска производственной деятельности для персонала, населения, окружающей среды осуществляют мониторинг, ограничения, защиту [4].

Управление техногенным риском, управление безопасностью профессиональной деятельности по большому счету сводится к разработке и реализации программ деятельности по предотвращению аварий, снижению их возможных последствий, обеспечению мониторинга, ограничений и защиты в процессе производственной деятельности [4].

В качестве примеров реальных мер, осуществляемых с целью управления техногенным риском, могут быть названы:

- мониторинг состояния объектов;
- прогнозирование чрезвычайных ситуаций и оценка их риска;
- предотвращение аварий и техногенных катастроф путем повышения технологической безопасности производственных процессов и эксплуатационной надежности оборудования;
- разработка и осуществление инженерно-технических мер по снижению возможных потерь и ущерба от чрезвычайных ситуаций;
- подготовка объектов экономики и систем жизнеобеспечения населения к работе в условиях чрезвычайных ситуаций;
- декларирование промышленной безопасности и лицензирование;
- проведение государственного надзора и контроля по вопросам и техногенной безопасности;
- страхование рисков и т.д.

Цель этого управления – достижение приемлемого уровня риска.

Выводы.

В системе управления угольным предприятием каждый уровень иерархии отвечает за выполнение своих целей, задач, функций управления, определяющихся в соответствии с общими целями и задачами компании. Одни структурные уровни компании связаны с другими информационными потоками, причем от качества этих потоков зависит функционирование всей системы управления.

Развитие и интенсификация промышленных производств в современных условиях неизбежно ведет к возрастанию числа аварий и масштабов последствий, связанных с неконтролируемым выбросом токсичных или взрывоопасных веществ в атмосферу. В связи с этим возникает необходимость использования научно-обоснованных подходов для обеспечения безопасности людей.

Литература

1. Ефремова, О.С. Охрана труда в организации в схемах и таблицах / О.С. Ефремова. – М.: Альфа-Пресс, 2012. – 108 с.
2. Ефремова, О.С. Охрана труда от А до Я: Практическое пособие / О.С. Ефремова. – М.: Альфа-Пресс, 2013. – 672 с.
3. Беляков, Г.И. Безопасность жизнедеятельности. Охрана труда: Учебник для бакалавров / Г.И. Беляков. – М.: Юрайт, 2013. – 572 с.
4. Методические указания по проведению анализа риска опасных производственных объектов РД 03–418–01 (утв. Постановлением Госгортехнадзора РФ от 10 июля 2001 г. N 30).
5. Хенли Э., Кумамото Х. Надежность технических систем и оценка риска. М., Горное дело, 1984.

УДК 004.891

ПРОГНОЗИРОВАНИЕ ОБЪЁМОВ ПРОДАЖ В EXCEL

Хоменко В.В., Орлов Ю.К.

Донецкий национальный технический университет
Кафедра искусственного интеллекта и системного анализа

E-mail: V_VKhomenko@mail.ru

Аннотация:

Хоменко В.В., Орлов Ю.К. Прогнозирование объёмов продаж в Excel. Анализ продаж, пакет прикладных программ Exel, прогноз, коэффициент сезонности, линейный тренд, график.

Annotation:

Khomenko V.V., Orlov Yu.K. Forecast sales in Exel. Analysis of sales, Sales analysis, software package exel, forecast, seasonality factor, linear trend, schedule.

Общая постановка проблемы

На сегодняшний день наука достаточно далеко продвинулась в разработке технологий прогнозирования. Специалистам хорошо известны методы нейросетевого прогнозирования, нечёткой логики и т.п. Разработаны соответствующие программные пакеты, но на практике они, к сожалению, не всегда доступны рядовому пользователю, а в то же время многие из этих проблем можно достаточно успешно решать, используя методы исследования операций, в частности имитационное моделирование, теорию игр, регрессионный и трендовый анализ, реализуя эти алгоритмы в широко известном и распространённом пакете прикладных программ MS Excel.[1-2] В данной статье представлен один из возможных алгоритмов построения прогноза объёма реализации для продуктов с сезонным характером продаж. Сразу следует отметить, что перечень таких товаров гораздо шире, чем это кажется. Понятие “сезон” в прогнозировании применим к любым систематическим колебаниям, например, если речь идёт об изучении товарооборота в течение недели под термином “сезон” понимается один день[4]. Кроме того, цикл колебаний может существенно отличаться (как в большую, так и в меньшую сторону) от величины один год. И если удаётся выявить величину цикла этих колебаний, то такой временной ряд можно использовать для прогнозирования с использованием аддитивных и мультипликативных моделей[3].

Исследование

Рассчитаем прогноз по продажам с учетом роста и сезонности. Проанализируем продажи за 12 месяцев предыдущего года и построим прогноз на 3 месяца следующего года с помощью линейного тренда. Каждый месяц это для нашего прогноза 1 период (y).

Уравнение линейного тренда:

$$y = bx + a,$$

где, y — объёмы продаж;

x — номер периода;

a — точка пересечения с осью y на графике (минимальный порог);

b — увеличение последующих значений временного ряда.

У нас имеются следующие статистические данные по продажам за прошлый год, которые приведены на рисунке 1.

	A	B	C
1	Месяц	Объём продаж (y)	Периоды (x)
2	Январь	17 986 229	1
3	Февраль	23 571 965	2
4	Март	25 537 589	3
5	Апрель	24 630 951	4
6	Май	24 429 696	5
7	Июнь	26 116 377	6
8	Июль	27 931 501	7
9	Август	25 914 893	8
10	Сентябрь	24 904 130	9
11	Октябрь	22 360 354	10
12	Ноябрь	23 825 299	11
13	Декабрь	22 241 744	12

Рисунок 1 - Статистические данные по продажам за прошлый год.

Рассчитаем значение линейного тренда. Определим коэффициенты уравнения $y = bx + a$. В ячейке D15 Используем функцию ЛИНЕЙН, которая приведена на рисунке 2.

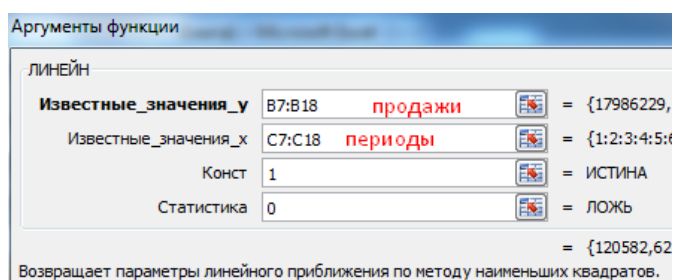


Рисунок 2 - Функция ЛИНЕЙН.

Выделяем ячейку с формулой D15 и соседнюю, правую, ячейку E15 так чтобы активной оставалась D15. Нажимаем кнопку F2. Затем Ctrl + Shift + Enter (чтобы ввести массив функций для обеих ячеек). Таким образом получаем сразу 2 значения коэффициентов для (a) и (b), которые приведены на рисунке 3.

D15	fx {=ЛИНЕЙН(B2:B13;C2:C13;1;0)}				
	A	B	C	D	E
14		линейный тренд: $y=bx+a$		b	a
15				120582,64	23337106,82

Рисунок 3 - Значения коэффициентов для (a) и (b).

Рассчитаем для каждого периода y-значение линейного тренда. Для этого в известное уравнение подставим рассчитанные коэффициенты (x – номер периода), которые видно на рисунке 3.

C	D	E
Периоды (x)	Значение тренда	
1	=D\$15*C2+\$E\$15	
2	23578272,10	
3	23698854,75	
4	23819437,39	
5	23940020,03	
6	24060602,68	
7	24181185,32	
8	24301767,97	
9	24422350,61	
10	24542933,25	
11	24663515,90	
12	24784098,54	
$y=bx+a$	b	a
	120582,64	23337106,82

Рисунок 3 – Подставление коэффициентов.

Чтобы определить коэффициенты сезонности, сначала найдем отклонение фактических данных от значений тренда («продажи за год» / «линейный тренд»), которые приведены на рисунке 4.

E	
Отклонение фактических продаж от значений тренда	
=B2/D2	
	0,999732504
	1,077587473
	1,034069386
	1,020454284
	1,085441514
	1,155092301
	1,066378917
	1,019726987
	0,91107097
	0,966013893
	0,897419931

Рисунок 4 - Отклонение фактических данных от значений тренда.

Рассчитаем средние продажи за год. С помощью формулы СРЗНАЧ.

Средние продажи за год	
24 120 894 <--	=СРЗНАЧ(B2:B13)

Рисунок 5 – Средние продажи за год.

Индекс сезонности для каждого месяца (отношение продаж месяца к средней величине) приведён на рисунке 6. Фактически каждый объем продаж за месяц делится на средний объем продаж за год.

G	
Индекс сезонности по месяцам	
=B2/\$F\$2	
	0,98
	1,06
	1,02
	1,01
	1,08
	1,16
	1,07
	1,03
	0,93
	0,99
	0,92

Рисунок 6 – Индекс сезонности по месяцам.

В ячейке H2 найдем общий индекс сезонности через функцию: =СРЗНАЧ(G2:G13).

Спрогнозируем продажи, учитывая рост объема и сезонность. На 3 месяца вперед. Продлеваем номера периодов временного ряда на 3 значения в столбце I. На рисунке 7 приведён период для прогноза.

I
Периоды для прогноза
13
14
15

Рисунок 7 – Период для прогноза.

Рассчитаем значения тренда для будущих периодов: изменим в уравнении линейной функции значение x. Для этого можно просто скопировать формулу из D2 в J2, J3, J4.

На основе полученных данных составляем прогноз по продажам на следующие 3 месяца (следующего года) с учетом сезонности и представим его на рисунке 8.

	G	H	I	J	K
	Индекс сезонности по месяцам	Общий индекс сезонности	Периоды для прогноза	Значение тренда на следующие 3 месяца	Прогноз на следующие 3 месяца с учетом сезонности
1					
2	0,75	1	13	24904681,18	=J2*G2
3	0,98		14	25025263,83	24455753,71
4	1,06		15	25145846,47	26622740,11
5	1,02				

Рисунок 8 - Прогноз по продажам с учетом сезонности.

Общая картина составленного прогноза представлена на рисунке 9.

Месяц	Объем продаж (y)	Периоды (x)	Значение тренда	Отклонение фактических продаж от значений тренда	Средние продажи за год	Индекс сезонности по месяцам	Общий индекс сезонности	Периоды для прогноза	Значение тренда на следующие 3 месяца	Прогноз на следующие 3 месяца с учетом сезонности
Январь	17 986 229	1	23457689,46	0,766751944	24 120 894	0,75	1	13	24904681,18	18570675,65
Февраль	23 571 965	2	23578272,10	0,999732504		0,98		14	25025263,83	24455753,71
Март	25 537 589	3	23698854,75	1,077587473		1,06		15	25145846,47	26622740,11
Апрель	24 630 951	4	23819437,39	1,034069386		1,02				
Май	24 429 696	5	23940020,03	1,020454284		1,01				
Июнь	26 116 377	6	24060602,68	1,085441514		1,08				
Июль	27 931 501	7	24181185,32	1,155092301		1,16				
Август	25 914 893	8	24301767,97	1,066378917		1,07				
Сентябрь	24 904 130	9	24422350,61	1,019726987		1,03				
Октябрь	22 360 354	10	24542933,25	0,91107097		0,93				
Ноябрь	23 825 299	11	24663515,90	0,966013893		0,99				
Декабрь	22 241 744	12	24784098,54	0,897419931		0,92				
линейный тренд:	$y=bx+a$	b	a							
		120582,64	23337106,82							

Рисунок 9 - Общая картина составленного прогноза.

График прогноза продаж представлен на рисунке 10.



Рисунок 10 - График прогноза продаж
График сезонности представлен на рисунке 11.

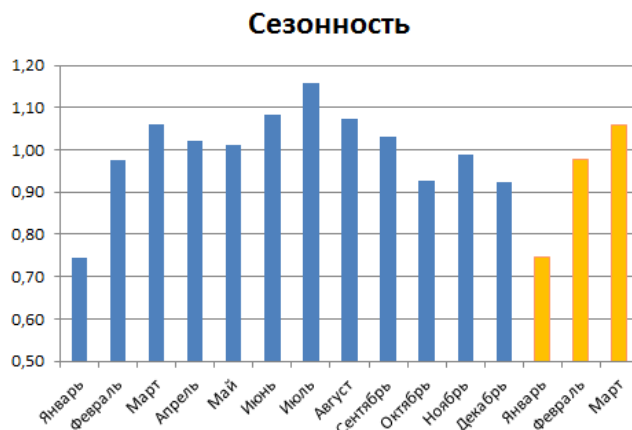


Рисунок 11 - График сезонности.

Выводы

Для обеспечения полного и своевременного прогнозирования потенциального спроса, издержек и цен необходимо использовать программы для анализа и прогнозирования, среди таких программы выделяют, например Пакет анализа MS Excel который обеспечивает не только простой ввод данных для анализа и прогнозирования, но и самым процесс анализа и прогнозирования, таким образом, обеспечивая формирование прогнозируемых показателей в наиболее приемлемой форме, обеспечивающей быстроту и правильность расчетов. Для создания электронной таблицы достаточно выполнить команду Пуск - Программы - MS Excel, после чего можно вводить данные. Построение функций(графиков) осуществляется благодаря команде Вставка - Функция (Диаграмма). На основании введенных данных можно осуществлять анализ, определять динамику, строить графики, также можно осуществлять их группировку и сортировку с помощью команды Данные - Фильтр - Автофильтр.

Литература

1. Баканов, М. И. Теория экономического анализа / М. И. Баканов. – Москва: Финансы и статистика, 2007. – 416 с.
2. Савицкая, Г. В. Анализ хозяйственной деятельности предприятия / Г. В. Савицкая. – Москва: Экоперспектива, 2005. – 704 с.
3. Галпин, Т. Системный анализ / Т. Галпин, М. Хэндон. – Москва: Вильямс, 2006.– 150 с.
4. Завьялова, Е., Системный анализ / Е. Завьялова, Е. Этокова. – Москва: Рант, 2005. – 140 с.

УДК 339.138

РАЗРАБОТКА МОДЕЛИ РАБОТЫ СИСТЕМЫ ПОТРЕБЛЕНИЯ ТОПЛИВНО-ЭНЕРГЕТИЧЕСКИХ РЕСУРСОВ

Орлов Ю.К., Чайка А.Р.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: oleg492000@mail.ru

Аннотация:

Орлов Ю.К., Чайка А.Р. Разработка модели работы системы потребления топливно-энергетических ресурсов. Описаны проблемы предприятия, описана существующая модель процесса работы объекта управления, описано организационно-экономическое обеспечение по эксплуатации модели объекта управления, предложена модель процесса работы объекта управления.

Annotation:

Orlov Yu.K., Chaika A.R. Development of a model of the system of consumption of fuel and energy resources. The problems of the enterprise are described, the existing model of the management object's work process is described, the organizational and economic support for the operation of the control object model is described, and the model of the control object's work process is proposed.

Общая постановка проблемы

Современное состояние проблемы в рассматриваемом объекте управления очень велико. Государственное предприятие «Главное управление статистики ДНР», а именно отдел статистики торговли использует устаревший процесс работы, а именно:

- принимают отчеты в бумажном носителе;
- предоставляют статические данные в выше стоящие органы в бумажном носителе.

Используют более устаревшее ПО, (программное обеспечение), а именно:

- используют для анализа документа более устаревшие ПО (Statistica);
- используют для введения базы данных более устаревшие ПО (Excel).

Исследование

Так, как главная проблема предприятия – это работа в устаревшем режиме. В моей работе я создала две модели, это работа в существующем режиме и работа в предложенном режиме.

Государственное предприятие «Главное управление статистики ДНР», а именно отдел статистики торговли использует устаревший процесс работы:

- принимают отчеты в бумажном носителе;
- предоставляют статические данные в выше стоящие органы в бумажном носителе.

Используют более устаревшее ПО, (программное обеспечение), а именно:

- используют для анализа документа более устаревшие ПО (Statistica);
- используют для введения базы данных более устаревшие ПО (Excel).

Этапы осуществления процесса работы предприятия в более комфортном режиме:

- 1) сбор отчетов от предприятий в электронном режиме;

- 2) использование более новой программы для анализа (Stata);
- 3) использование более новой программы для введения базы данных (Navisat);
- 4) предоставление статической информации в выше стоящие органы в электронном режиме.

Существующая модель объекта управления

Существующая модель системы управления объектом изображена на рисунке 1.

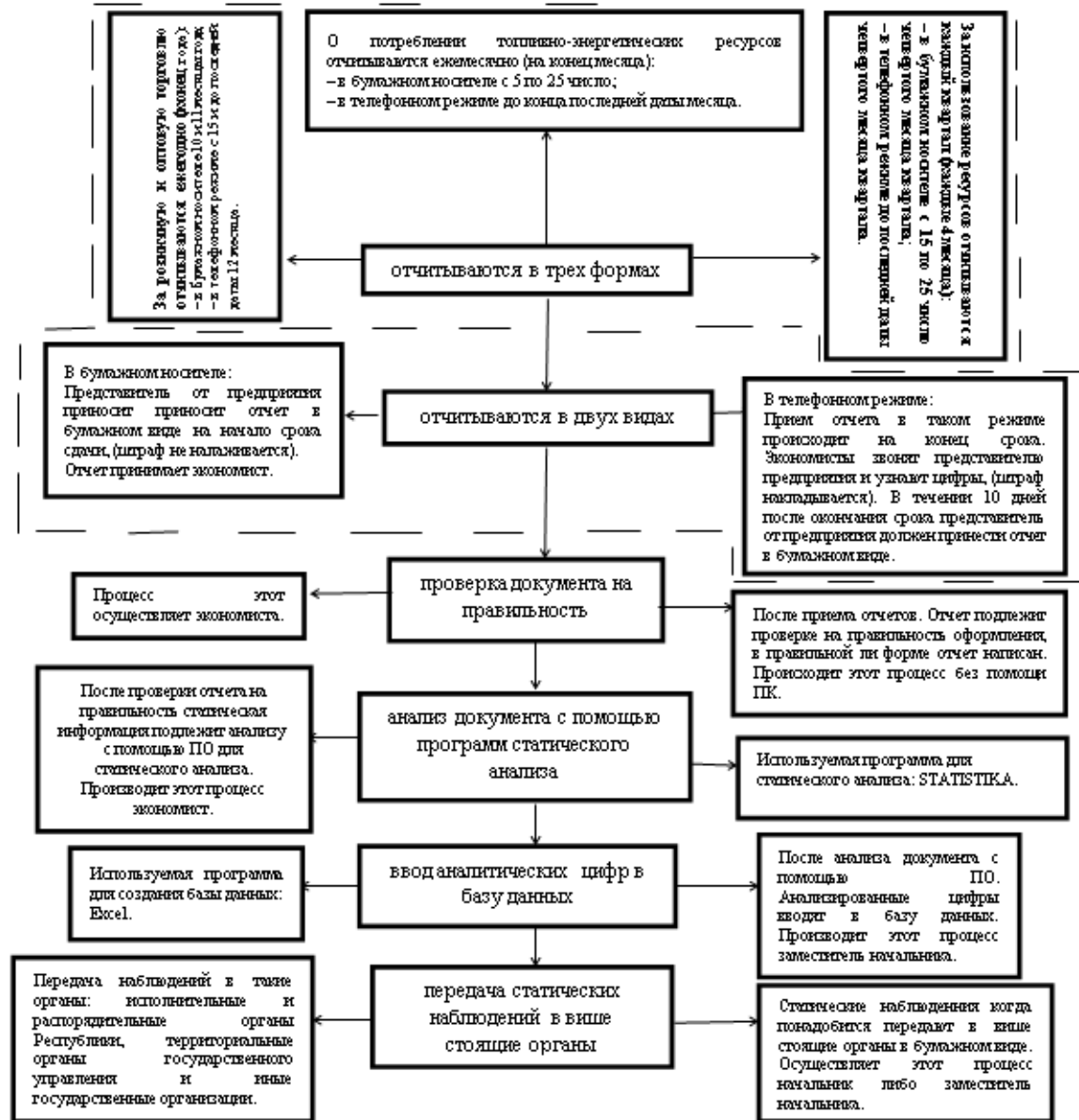


Рисунок 1 – Существующая модель системы управления «Отдел статистики торговли»

Предложенная модель для системы объекта управления изображена на рисунке 2.

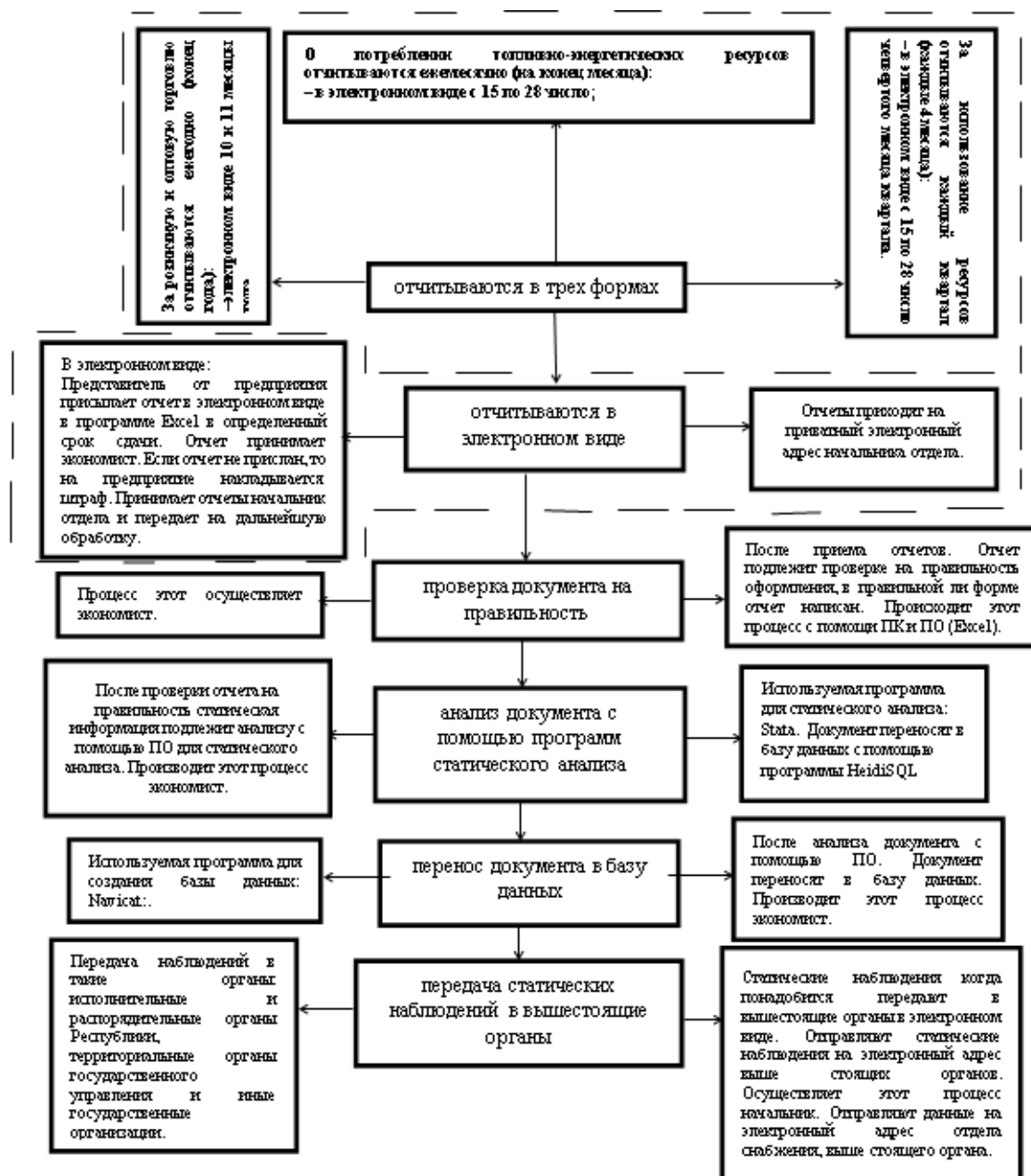


Рисунок 2 – Предложенная модель для системы объекта управления «Отдел статистики торговли»

Элементы новизны

Была выбрана модель усовершенствования системы для облегчения работы сотрудников отдела. Необходимо внести некоторые изменения в процесс работы отдела, а именно:

- 1) осуществлять сбор отчетов от предприятий в электронном режиме;
- 2) использовать более новую программу (STATA) для статического анализа;
- 3) использовать более новую программу (Navicat) для создания и введения базы данных;
- 4) использовать программу (HeidiSQL) для переноса анализированных цифр в базу данных;

5) осуществлять предоставление статической информации высшим органам в электронном виде.

Оценка экономической эффективности предложенной модели для объекта управления

Затраты на подготовку производства: можно выделить финансы на закупку лицензированных программ с помощью которых будут создаваться статические наблюдения. И приватные хранилища в виде внутренних накопителей жестких дисков с достаточным объемом памяти. Рабочие лицензированные аккаунты систем для выполнения отчетности, а также аккаунты электронных почт сотрудникам на которые будут отсылаться отчеты различных ведомств.

Затраты на реализацию незначительные: денежные средства несколько десятков тысяч рублей, плюс время на перевод персонала на новые системы автоматизированных систем, их обучение, переквалификация, внедрение.

Необходимость в части требования рынка: введение более современных способов отчетности в различных структурах, где есть какие-либо отчетности.

Выводы

Мной выбрана изменить именно эти детали потому, что процесс с этими измененными деталями будет более эффективный и менее ресурсно- затратный, более экономный во времени и позволит увеличить эффективность, работоспособность отдела. Усовершенствование систем управления статических отчетов позволит выйти на более продвинутый Европейский уровень, если начать использовать электронные методы отчетности. Например, такие как почта в формате vnp. Сдача отчетов в электронном виде позволит с экономить время обработки статических данных, а также увеличить план отчетностей.

Полезное: снижение ресурсно-затратности по времени, плюс снижения нагрузки персонала и избавления от ненужных пустых затрат.

Предложенную модель, а именно осуществление работы в электронном режиме, использовать можно именно на Главном управлении статистики ДНР, но использовать можно не только в рассматриваемом отделе статистики торговли, а на всех подобных отделах государственного предприятия.

Литература

1. Кожевникова, И.А. Стохастическое моделирование процессов / И.А. Кожевникова – Чебоксары: Чувашский государственный университет имени И. Н. Ульянова, 2018. – 146 с.
2. Малугин, В.А. Теория вероятностей и математическая статистика / В.А. Малугин – М.: Московский государственный университет имени М.В. Ломоносова, 2018. – 38 с.
3. Мхитарян, В.С. Анализ данных / В.С. Мхитарян – М.: Национальный исследовательский университет «Высшая школа экономики», 2005. – 678 с.

УДК 303.732.4

МОДЕЛИРОВАНИЕ ПРОЦЕССА КРИСТАЛЛИЗАЦИИ МЕТАЛЛА В ЭЛЕКТРОШЛАКОВОМ ПЕРЕПЛАВЕ

Миненко А.С., Шихов И.В.

Донецкий национальный технический университет
кафедра автоматизированных систем управления

E-mail: ivan.shihov2012@gmail.com

Аннотация:

Миненко А.С., Шихов И.В. Моделирование процесса кристаллизации металла в электрошлаковом переплаве. Рассмотрена модель кристаллизации металла в электрошлаковом переплаве. Определен подход формирования требований к модели и оборудованию для вытягивания слитка при электрошлаковом переплаве.

Annotation:

Minenko A.S., Shikhov I.V. Modeling the process of metal crystallization in electroslag remelting. A model of metal crystallization in electroslag remelting is considered. The approach to the formation of requirements for the model and equipment for pulling the ingot during electroslag remelting has been determined.

Общая постановка проблемы

Проблема построения математических моделей в области электрометаллургии является актуальной, так как изучение некоторых объектов экспериментальным путем не представляется возможным ввиду их сложного процесса функционирования.

Исследования

Пусть $D = (-1 < x < 1, y < 0)$ полуполоса, заполненная твердым металлом. Обозначим через $u(x, y)$ температуру этого металла. Требуется определить температуру $u(x, y)$ по следующим условиям:

$$\begin{aligned}u_{xx} + u_{yy} + \omega u_y &= 0, (x, y) \in D, & (1) \\u_x \pm \omega_0 u &= 0, x = \pm 1, -\omega < y < 0, & (2) \\u(x, -\infty) &= 0, & (3) \\u_y(x, 0) &= v(x), -1 \leq x \leq 1, & (4)\end{aligned}$$

где ω – постоянная, число Пекле, характеризующее соотношение между процессами переноса тепла в потоке жидкости (соотношение конвекции и диффузии);

ω_0 – постоянная, число Нуссельта, характеризующее соотношение между интенсивностью теплообмена за счет конвекции и интенсивностью теплообмена за счет теплопроводности.

Выполним преобразования формулы (1) и придем к системе дифференциальных уравнений:

$$\begin{aligned}\frac{X''Y}{XY} + \frac{Y''X}{XY} + \omega \frac{Y'X}{XY} &= 0; \\ \frac{X''}{X} &= -\frac{Y''}{Y} - \omega \frac{Y'}{Y} = -\lambda^2;\end{aligned}$$

$$\begin{cases} X'' + \lambda^2 X = 0 \\ Y'' + \omega Y' = \lambda^2 Y, \end{cases}$$

где λ – коэффициент теплопроводности.

Решая первое дифференциальное уравнение, получим следующий результат
 $X = C_1 \cos \lambda x + C_2 \sin \lambda x$.

Согласно условию (2) и учитывая, что $u = X(x)$, получим

$$\begin{aligned} X' \pm \omega_0 X &= 0; \\ -\lambda(\cos \lambda x - \sin \lambda x) \pm \omega_0(\cos \lambda x + \sin \lambda x); \\ \omega_0 &= \pm \lambda \frac{\cos \lambda x - \sin \lambda x}{\cos \lambda x + \sin \lambda x} = \pm \lambda \frac{\cos^2 \lambda x - \sin^2 \lambda x}{2 \sin \lambda x \cos \lambda x} = \pm \lambda \frac{\cos 2\lambda x}{\sin 2\lambda x} = \pm \lambda \operatorname{ctg} 2\lambda x. \end{aligned}$$

Решая второе дифференциальное уравнение из вышеприведенной системы, и учитывая, что $Y(y) = e^{\mu y}$, получим:

$$\begin{aligned} Y'' + \omega Y' &= \lambda^2 Y; \\ \mu^2 + \omega \mu - \lambda^2 &= 0; \\ \mu_n &= \frac{-\omega \pm \sqrt{\omega^2 + 4\lambda_n^2}}{2} = -\frac{\omega}{2} \pm \sqrt{\frac{\omega^2}{4} + \lambda_n^2}. \end{aligned}$$

Таким образом, решение задачи (1) – (4) имеет вид

$$\begin{aligned} u_n(x, y) &= 2v_n \frac{\cos \lambda_0 x \sin \lambda_0 (1 - e^{\lambda_0 H})}{\lambda_0 \mu_0 (1 + \omega_0 \frac{\cos^2 \lambda_0}{\lambda_0^2})} e^{\mu_0 y} + \\ &+ 2\omega_0 v_n \sum_{k=1}^n \frac{\cos \lambda_0 x \cos \lambda_k (1 - e^{\mu_k H})}{\lambda_k^2 \mu_n (1 + \omega_0 \frac{\cos^2 \lambda_k}{\lambda_k^2})} e^{\mu_k y}, \end{aligned} \quad (5)$$

где $\mu = -\frac{\omega}{2} \pm \sqrt{\frac{\omega^2}{4} + \lambda_n^2}$, $n = 1, 2, 3, \dots, \lambda_n$ – положительные корни уравнения $\lambda = \omega_0 \operatorname{ctg} \lambda x$.

Отождествим теперь температуру $u(x, y)$ с температурой твердого слитка, находящегося в кристаллизаторе при электрошлаковом переплаве.

Для вытягивания слитка из кристаллизатора поверхность слитка предварительно обогревается тремя электронными лучами W_1, W_2 и W_3 , причем мощность W_3 одного из них равномерно распределена в центральной зоне $\{-1 \leq x \leq 1, y = 0\}$, а два других сконцентрированы по краям $x = \pm 1$.

Независимо от того, в каком отношении находится температура поверхности слитка с критической температурой T^k , при которой поверхность слитка отделяется от стенок кристаллизатора, теплообмен слитка с кристаллизатором осуществляется по формуле (2).

Для получения температуры слитка достаточно положить в формуле (5), что $v(x) = (W_1, W_2, W_3)$.

Далее введем в рассмотрение функционал:

$$I(v) = \int_H^0 (u(1,y) - T^k)^2 dy, \quad (6)$$

где H – высота слитка.

Требуется определить поток $v(x)$ из допустимого множества U , доставляющий наименьшее значение функционалу $I(v)$.

При численной реализации задачи необходимо учесть следующее ограничение $2500 \leq v(x) \leq 5000$, $v(x)$ – мощность потока в единицах МВт/м².

Выводы

При электрошлаковом переплаве, плавления электрода и кристаллизация расплавленного металла происходят одновременно. В каждый момент времени существует затвердевшая часть слитка в чашеобразной форме ванна жидкого металла, покрыта сверху толстым слоем расплавленного шлака. Над ванной жидкого металла образуется более плотный слиток, чем в случае дугового переплава.

При электрошлаковой плавке в медном водоохлаждающем кристаллизаторе создается направленность роста кристаллов слитка снизу вверх. В связи с этим в слитке, отлитым таким способом, полностью отсутствуют разного рода дефекты, образующиеся при обычной кристаллизации жидкого металла в изложнице: усадочные раковины, осевая рыхлость и осевая кристаллизационные трещины. Отсутствует также зональная сегрегация элементов. Отсутствие дефектов способствует улучшению его пластичности при высоко-температурной механической обработке (прошивке, ковке, прокатке и т. д.). Наличие тонкой корочки шлака между слитком и внутренней стенкой кристаллизатора обуславливает формирование гладкой и ровной поверхности слитков, что позволяет ковать или прокатывать их без предварительной обдирки.

Данные формулы позволяют определить температуру, при которой слиток необходимо вытянуть из кристаллизатора.

Литература

1. Миненко, А.С. Вариационные задачи со свободной границей / А.С. Миненко – К.: Наук. Думка, 2005. – 341 с.
2. Дакурорт, У. Электрошлаковый переплав / У. Дакурорт, Д. Хойл; [пер. с англ.]. – М.: Металлургия, 1973. – 192 с.
3. Шевченко, А.И. Моделирование одного класса сложных с нечеткими управлениями / А.И. Шевченко, А.С. Миненко, И.А. Сыпко // Доп. НАН Украины. – №8. – 2013. – С. 52-54.
4. Грибан, В.Г. Охрана труда : учебное пособие для студентов высших учебных заведений / В.Г. Грибан, О.В. Негодченко. – К.: Центр учебной литературы, 2009. – 280 с.
5. Городецкий, В.В. Методы решения задач по функциональному анализу: Учебное пособие / В.В. Городецкий, Н.И. Нагнибида, П.П. Настасиев. – К.: Вища шк., 1990. – 479 с.
6. Матвеев, Н.М. Методы интегрирования обыкновенных дифференциальных уравнений / Матвеев Н.М. – М.: Высшая школа, 1967. – 564 с.
7. Эльсгольц, Л.Э. Дифференциальные уравнения и вариационное исчисление / Эльсгольц Л.Э. – М.: Наука, 1969. – 424 с.
8. Клюев, М.М. Электрошлаковый переплав / М.М. Клюев, С.Е. Волков. – М.: Металлургия, 1984. – 208 с.
9. Шевченко, А.И. Численный анализ одной нелинейной математической модели / А.И. Шевченко, А.С. Миненко, О.А. Золотухина // Доп. НАН Украины. – №10. – 2012. – С. 35-38.

УДК 004.891.2

РАЗРАБОТКА СИСТЕМЫ ПЛАНИРОВАНИЯ ЗАКУПОК ТОВАРОВ МЕДИЦИНСКОГО НАЗНАЧЕНИЯ НА КОММЕРЧЕСКОМ ПРЕДПРИЯТИИ

Удовика Ю.Д., Тарасова И.А.

Донецкий национальный технический университет, г. Донецк
кафедра искусственного интеллекта и системного анализа

E-mail: youlia.u@mail.ru

Аннотация:

Удовика Ю.Д., Тарасова И.А. Разработка системы планирования закупок товаров медицинского назначения на коммерческом предприятии. В работе рассмотрены особенности планирования закупок товаров медицинского назначения, разработаны методы и алгоритмы системы планирования закупок, выполнена программная реализация системы.

Annotation:

Udovika J.D., Tarasova I.A. Development of a planning system for the procurement of medical products at a commercial enterprise. The paper discusses the features of planning the procurement of medical supplies, developed methods and algorithms for the procurement planning system, implemented a software implementation of the system.

Постановка проблемы

Планирование закупок на непроизводственном коммерческом предприятии целесообразно в тех случаях, когда продаваемая продукция имеет сезонный спрос и узкую торговую направленность. При этом деятельность предприятия должна вестись не менее трех лет – это позволяет установить предпочтения покупателей и сезонность продаваемой продукции – востребованность в определенный временной промежуток. Данный показатель зависит от множества факторов, обычно не учитываемых при информационном учете на предприятии. Примерами таких факторов в области продажи товаров медицинского назначения могут быть регулярные медицинские профилактические осмотры в государственных организациях или статистика заболеваемости в регионе. Так же необходимо учитывать специфику хранения и перевозок медицинских изделий (например: температурный режим). В основном существующие программные продукты, специализирующиеся на планировании закупок, не учитывают данные факторы или требуют дорогостоящей настройки под конкретную область деятельности коммерческого предприятия [1].

Цель статьи – минимизировать время на принятие решений в области логистической деятельности в сфере закупок коммерческого предприятия за счет разработки системы планирования закупок товаров медицинского назначения.

Для достижения поставленной цели в работе выполнено:

- 1) исследование предметной области;
- 2) разработка методов системы планирования закупок;
- 3) разработка алгоритма распределения товаров при транспортировке;
- 4) программная реализация системы планирования закупок.

Исследование предметной области

Товары (изделия) медицинского назначения – это медицинские изделия из стекла, полимерных, резиновых, текстильных и иных материалов, наборы реагентов и контрольные материалы для них, другие расходные средства и изделия, в основном однократного применения, не требующие технического обслуживания при использовании. Они условно подразделяются по нескольким характеристикам: температурные условия хранения,

временные условия хранения и занимаемое место (объем, площадь или вес) при их транспортировке. Информация о каждом номенклатурном элементе или группе единиц товаров и их характеристиках хранится в базе данных. Существующие системы планирования, такие как MRP-1/MRP-2, KANBAN, JIT, SDP, LP, LRP и другие удовлетворяют требованиям производственных предприятий, а не торговых. Используемые наиболее популярными системами планирования аналитические методы при их комбинировании позволяют рассчитывать объем необходимых для продажи ресурсов, следовательно – формируют необходимый минимум для закупки. Но эти данные не содержат всей информации о сфере деятельности коммерческого предприятия и не учитывают всех факторов, влияющих на торговлю.

Программный комплекс, используемый для учета движения товаров на рассматриваемом коммерческом предприятии – это 1С.Предприятие v8.3. Проводимый на протяжении трех лет информационный и бухгалтерский учет в данном программном продукте, а также гибкость системы позволяет разработать дополнительную конфигурацию, специализирующуюся на планировании закупок. Основная информация хранится в базах данных товаров и выводится из расходных и приходных накладных и отчетов бухгалтерии.

Разработка методов системы планирования закупок

Основным показателем необходимым для планирования закупок является коэффициент сезонности – частота заказов конкретного изделия, используемая как множитель при расчете количества закупаемых товаров. Для его выведения необходимо проанализировать продажи и вывести периоды с наибольшей и наименьшей потребностью в товаре по всем или по конкретному контрагенту за прошедшие периоды (от месяца до года). Для этого лучше применить совмещенный ABC/XYZ анализ, где основным методом разбиения товаров на группы является метод простой средней. Алгоритм метода следующий:

Шаг 1. Вычисление среднего уровня продаж в год, как отношение общего числа проданных товаров за год к количеству рабочих месяцев.

Шаг 2. Вычисление промежуточного коэффициента сезонности в каждом месяце в течение года, как отношение количества проданных товаров в данном месяце к значению среднего уровня продаж в год (расчет необходимо вести для каждого года работы предприятия).

Шаг 3. Расчет среднеарифметического коэффициента для каждого месяца за все года работы предприятия.

Результаты анализа формируют базу периодов с выведенными коэффициентами сезонности для каждой номенклатурной единицы товаров и будут использоваться для дальнейшего планирования закупок. База обновляется пользователем и рассчитывается для заданного периода, так же в ней учитываются температурные характеристики товаров. Учитывая особенности деятельности предприятия, рекомендуется вести расчеты для каждого месяца, это связано с длительностью транспортировки товаров, частотой и объемом заказов. Объем, необходимых для закупки товаров, рассчитывается как разность между планируемыми продажами (здесь учитывается сезонность) и запасом товаров на складе (с учетом срока годности продаваемых изделий). Конечным результатом планирования является отчет с рекомендованным количеством закупки и сформированный план заказов.

Разработка алгоритма распределения товаров при транспортировке

При сформированном плане заказов существует необходимость транспортировки изделий, с учетом их объема, веса и занимаемой площади. Алгоритм должен определять количество необходимых транспортных средств и учитывать приоритетность доставки товаров. Приоритетность – это критерий, определяемый пользователем (менеджером отдела закупки). Транспорт по грузоподъемности делится на: малый (0,5-2 тонны), средний (от 2 до 5 тонн), большой (5-16 тонн), особо большой (от 16 тонн). Алгоритм распределения товаров по машинам следующий:

Шаг 1. Определение приоритетности и веса товаров в очереди.

Шаг 2. Нахождение минимальной единицы товара (по площади/объему или весу) и разбиение остального груза на полученную минимальную единицу.

Шаг 3. Определение количества необходимых машин с учетом бюджета закупки.

Программная реализация системы планирования закупок

Основная полученная информация представляется в виде отчетов и диаграмм. Пользователь имеет возможность группировать выходную информацию, выбирая контрагента или наименование товара медицинского назначения (см. рис. 1). Результаты промежуточной работы такие как отчет по ABC/XYZ анализу или графическое отображение частоты покупаемых товаров определенным контрагентом также отображается в отчетах и доступно для бухгалтерского или информационного учета (см. рис. 2).

Номенклатура	Продано	Дней продаж	Средняя продажа за день	Необходимый объем закупки	Планируемый объем продаж
Игла S-Monovette 21 С 100 штуп	2	1	2	2	2
Колба Каледалье 1-250-19/26	1	1	1	1	1
Масло иммерсионное 100мл	2	1	2	2	2
Пилетка Пастера diam 6 мм дл 18 см уп 50шт/200шт	50	1	50	50	50
Пробирка центрифужная П1-10 (ПЦНГ), МиниМед	300	1	300	300	300
Пробирка Вакуумная с желто-оранжевой на диске сыворотки, с активатором сверт и гелем 8 мл, 15x100мм	500	1	500	500	500
Пробирка лабораторная ПБ2-16x150, МиниМед	100	1	100	100	100
Стекло покровное 18x18, 100шт	10	1	10	10	10
Термометр ТТЖ-М исп 1	10	1	10	10	10
Итого	975	9	975	975	975

Рисунок 1 – Пример планируемых закупок на неделю

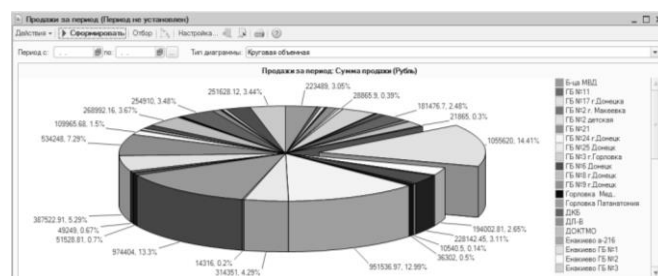


Рисунок 2 – Диаграмма продаж (сформированная за весь период)

Выводы

Анализ процесса закупки товаров любого предприятия учитывает множество факторов, влияющих на перевозку и заказ продукции. Транспортировка товаров медицинского назначения характеризуется особыми условиями хранения товаров. Некоторые номенклатурные единицы имеют хрупкое строение или температурные показатели, которые необходимо учитывать вместе с сезонностью закупок. Обычно процесс закупки товаров не автоматизирован на коммерческом предприятии, так как это ведет за собой достаточно большой объем систематизации номенклатурных единиц и учета множества факторов, а существующие программные комплексы требуют определенной настройки под предметную область торговой деятельности и рассчитываются в основном на предприятия производственного типа, а не торгового. Автоматизация процесса закупок реализована в программном комплексе, при помощи которого ведется торговая деятельность предприятия – 1С.Предприятие v8.3, позволяющем внедрять программные дополнения без экспорта баз данных.

Литература

1. Удовика Ю.Д. Анализ существующих систем планирования ресурсов и их методов работы для коммерческого предприятия / Ю.Д. Удовика, И.А. Тарасова, В.И. Елисеев // Сборник материалов студенческой секции IX Международной научно-технической конференции «Информатика, управляющие системы, математическое и компьютерное моделирование». 2018 г., Донецк, ДонНТУ – С. 447-449.

УДК 336.76

МЕТОДЫ ПРОГНОЗИРОВАНИЯ И АНАЛИЗ ФОНДОВОГО РЫНКА

Колесников В.Ю., Ванжа Т.В.

ГОУ ВПО «Донецкий национальный технический университет»
кафедра искусственного интеллекта и системного анализа

E-mail: Delfics02@yandex.ru

Аннотация:

Колесников В.Ю., Ванжа Т.В. Методы прогнозирования и анализ фондового рынка. Анализ методов, которые используются при прогнозировании котировок фондового рынка, от классических методов, используемых фундаменталистами и аналитиками-аналитиками, до новейших методов. Практические ситуации, где анализируемые методы являются подходящими, также указаны.

Annotation:

Kolesnikov V.Yu., Vanzha T.V. Forecasting methods and stock market analysis. Analysis of methods used in forecasting stock market quotes, from the classic methods, used by the fundamentalists and chartist analysts, to the newest methods. The practical situations, where the analyzed methods are suitable, are also indicated.

Общая постановка проблемы

Развитие фондового рынка является необходимостью, как для обеспечения непрерывного экономического роста, так и для эффективного распределения ресурсов в экономике. Основную роль в развитии фондового рынка играет законодательство, главным образом, в отношении конкретного закона, касающегося прозрачности, защиты и равного отношения к инвесторам. Он поддерживает доверие инвесторов на фондовом рынке и убеждает их инвестировать в него. Фондовый рынок выполняет различные функции, такие как:

- оптимальная структура для рыночных сделок;
- прозрачность сделок и защита инвесторов;
- генерация правильных цен;
- гармонизация цен для всей экономики;
- разделение и покрытие риска, среди прочего.

Котировки акций представляют цены сделок на биржевом рынке. Они отражают взаимосвязь между спросом и предложением на этом рынке. Основными факторами, которые влияют на котировки акций, являются:

- экономический статус получателя;
- ожидания инвесторов относительно прибыльности и дивидендов;
- эволюция национального и международного фондового рынка, и специфические особенности деятельности фондового рынка, которые влияют на спрос и предложение;
- политические, военные, культурные факторы и другие.

Анализ методов и выбор оптимального метода, который лучше подходит для конкретной ситуации и полностью оценивает существующие данные, имеет чрезвычайно важное значение.

Установленная на рынке цена становится инструментом экономического анализа и прогнозирования. Это наиболее реалистичный инструмент прогноза, используемый при принятии решения об экономической политике, и это сигнал к действию для экономических агентов правительства и местных органов власти. Именно поэтому прогноз котировок акций крайне важен.

Исследования

В зависимости от того, как оцениваются акции, аналитики фондового рынка могут быть классифицированы как:

- фундаменталисты, они рассматривают только фундаментальные факторы рынка;
- аналитики используют методы графического анализа.

Фундаментальный анализ основан на изучении экономики, поля и состояния общества с целью определения стоимости доли конкретной компании. Фундаментальный анализ отслеживает прибыль компании и дивиденды, которые компания предлагает, учитывает ожидания относительно процентных ставок и оценивает риск, связанный с компанией. Он использует статистические, математические и финансовые алгоритмы, применяемые к официальной периодической финансовой отчетности компании, чтобы как можно точнее оценить цену акций.

Технический анализ основан исключительно на изучении внутренних данных фондового рынка, учитывая, что все экономические, финансовые, политические и психологические факторы объединены в единый элемент: котировка акций. Технические аналитики изучают краткосрочные изменения цены акций, начиная с изучения истории котировок, с интервалом не менее 6 месяцев и предполагают, что поведение в прошлом будет распространяться на будущее. Технический анализ предлагает информацию о возможной будущей эволюции фондового рынка.

Теория эффективного рынка объясняет формирования цены акций, основанные на общей модели поведения. Так же важно понимать, что рынки не имеют памяти, то есть любая информация об акции мгновенно корректируется в цене. Вот почему прошлое не помогает предсказывать будущее. Основная часть пользователей это спекулянты и долгосрочные инвесторы. Слабая гипотеза говорит им, что исторические цены на акции являются случайными и не содержат полезной информации, которая должна приводить к превышению среднего дохода. Промежуточная гипотеза указывает на временную задержку корректировки цены, на новую информацию. Сильная гипотеза показывает, что никто не заработает больше, независимо от используемой стратегии, потому что вся информация находится внутри цены. Методы, которые используются пользователями это статистические методы (регрессионный и корреляционный анализ), проверка признаков, эконометрический анализ.

В последние несколько десятилетий была разработана новая теория эволюции цен на акции, а именно теория эффективного рынка. Основными атрибутами этой новой теории являются:

- рыночные цены акций отражают мгновенно и полностью всю соответствующую информацию, доступную в данный момент времени;
- цены акций должны всегда отражать их реальную ценность;
- информация должна быть недорогой или даже бесплатной, и должен быть доступен всем участникам рынка;
- цены на акции следуют эволюции случайного блуждания.

Вообще говоря, модель временного ряда будет предпочтительнее, когда у нас мало информации о факторах, влияющих на поведение переменной и у нас большое количество данных или основная цель короткая прогноз, то анализ временных рядов начинается с построения модели данных, свойства которой аналогичны генерации анализируемого процесса. Если предположить, что свойства анализируемого процесса, включенные в модель, сохраняются в будущем, то модель можно использовать для прогнозирования.

Экстраполяция с временными рядами отличается от простой экстраполяции. Разница возникает из-за того, что анализ временных рядов предполагает, что ряды, поведение которых должно быть предсказано, были созданы случайным образом, структуру которого можно охарактеризовать и описать. Другими словами, модель временных рядов дает описание природы (случайного) процесса, который генерировал временные ряды. Описание выполняется не с точки зрения причинно-следственной связи, как в случае регрессионной модели, а с точки зрения формы, в которой событие включено в процесс.

Модели временных рядов могут быть детерминированными или стохастическими, а модели, которые не ссылаются на источник или случайное изменение ряда, являются детерминированными. Стохастическая модель временного ряда даст больше информации в качестве детерминированной модели, позволяющей улучшить прогноз.

Традиционные методы анализа временных рядов предполагают, что ряд состоит из четырех элементов: тенденции, циклического компонента, сезонного компонента и случайных изменений. Первые три из вышеупомянутых компонентов являются детерминированными, систематическими, тогда как последний является остаточным компонентом, который обеспечивает анализируемые явления особенностью случайного процесса.

Роли компонентов в процессе прогнозирования различны в зависимости от продолжительности временного интервала, для которого выполняется прогноз. В краткосрочных прогнозах, остаточный компонент имеет большое значение. В долгосрочных прогнозах наиболее важным является трендовый компонент.

Широко используемый метод прогнозирования – экспоненциальное сглаживание. Простой вариант экспоненциального сглаживания подходит для рядов, которые не имеют явной тенденции. Метод Хольта-Винтерса подходит для серии с сильной тенденцией.

Методы скользящего среднего учитывают самые последние данные динамического ряда. Влияние последних данных уменьшается с увеличением количества используемых значений (периодов). Если динамический ряд имеет случайные изменения на больших временных интервалах, будет использоваться больший объем данных. Если динамический ряд имеет определенную конфигурацию и случайные изменения являются резкими и происходят через небольшие промежутки времени, то будет использовано меньшее количество данных.

Модели скользящих средних, безусловно, полезны, но они не дают информации о доверии к прогнозу и не объясняют случайное поведение временных рядов. Поэтому стохастические модели необходимы, потому что случайный компонент приносит информацию об ошибках прогноза.

Основой стохастического моделирования является случайный процесс. Можно предположить, что случайный процесс порождает временные ряды. Процесс состоит из упорядоченного набора случайных величин, связанных с вероятностными распределениями, определенными для каждого из t моментов времени. Мы считаем, что ряд ежедневных котировок акций генерируется дискретным стохастическим процессом реальных ценностей.

Простейший случайный процесс, который генерирует чисто случайные ряды, известен как белый шум и состоит из последовательности случайных независимых переменных, одинаково распределенных и равных нулю. Ряд – это белый шум, если он не имеет известной структуры (модели), и поэтому он непредсказуем.

Процесс случайного блуждания часто используется в качестве модели для котировок фондового рынка. Для этой конкретной модели все прогнозы равны последнему наблюдаемому значению, а доверительные интервалы выше по мере расширения прогнозируемого горизонта. Конкретная версия этого процесса, случайное блуждание с тенденцией, учитывает наличие тенденции и позволяет включить эту тенденцию в прогноз.

Существует много случайных процессов, для которых невозможно построить универсально допустимую модель с помощью скользящей средней или чисто авторегрессионной модели. Благодаря интеграции моделей авторегрессии и скользящего среднего была получена модель ARMA (смешанные модели авторегрессии - скользящего среднего). Это считается приемлемым по экономическим прогнозам, когда эволюция экзогенных перемен может быть неизвестна. Идентификация моделей ARMA основана как на функциях автокорреляции и частичной автокорреляции, так и на информационных критериях, используемых некоторыми специализированными программами прогнозирования.

Один из наиболее часто используемых и точных методов анализа и краткосрочного прогнозирования известен как метод Бокса-Дженкинса, основанный на концепции процесса ARIMA (серия интегрированных смешанных авторегрессий - скользящих средних).

Модель ARMA (p, q) подходит для моделирования стационарных процессов. Стационарный процесс имеет механизм генерации процессов, который инвариантен во времени. Среднее значение и дисперсия стационарного процесса не меняются во времени, а ковариация переменных зависит только от длины интервала времени, который разделяет две переменные. Трендовые и сезонные компоненты не встречаются в стационарных рядах. Нестационарные модели ARIMA (p, d, q) являются специфическими для несезонных явлений, тенденция которых может быть устранена, и, таким образом, процесс может быть сделан стационарным, благодаря определенным различиям определенного порядка d. Этапы построения модели ARIMA: идентификация модели, оценка параметров, проверка выбранной модели, использование модели для прогноза.

Выводы

Существуют значительные различия между методами прогнозирования с точки зрения их сложности, ограничений, требований и точности. Каждый метод подходит в четко определенных обстоятельствах. Данные методы должны присутствовать в любом исследовании эффективности, касающемся решения об инвестировании в акции.

Прогнозная деятельность, выполняемая либо экспертами, либо неспециалистами, почти всегда автоматизирована. Недавнее развитие прогнозирования и его возрастающая роль как фундаментальной основы для принятия решений во всех областях способствовали огромному развитию программного обеспечения для прогнозирования. Диапазон программного обеспечения, используемого для прогнозирования, простирается от программного обеспечения общего назначения, такого как электронные таблицы, до специализированного программного обеспечения, предназначенного исключительно для прогнозирования.

Литература

1. Егорова, Н.Е. Прогнозирование фондовых рынков с использованием экономико-математических моделей / Н.Е. Егорова, А.Р. Бахтизин, К.А. Торжевский. – М.: Красанд, 2013. – 216 с.
2. Дорохов, Е. В. Статистический анализ и прогнозирование развития фондового рынка России: дис. ... канд. экон. наук. – Москва. 2004. – 192 с.
3. Тертышный, С. А. Рынок ценных бумаг и методы его анализа. – П.: Питер, 2004. – 251 с.

4. Колби, Р. Энциклопедия технических индикаторов рынка. – Л.: Альпина Паблицер, 2013. – 837 с.
5. Акелис, С. Технический анализ от А до Я. – Л.: Омега-Л, 2010. – 376 с.

УДК 004.912

РАЗРАБОТКА СИСТЕМЫ ПОСТРОЕНИЯ ДЕРЕВА СИНТАКСИЧЕСКОГО ПОДЧИНЕНИЯ ДЛЯ АНАЛИЗА РУССКОЯЗЫЧНЫХ ТЕКСТОВ

Лёвкин А.В., Тарасова И.А.

Донецкий национальный технический университет
кафедра искусственного интеллекта и системного анализа

E-mail: a.lewckin2010@yandex.ru

Аннотация:

Лёвкин А.В., Тарасова И.А. Разработка системы построения дерева синтаксического подчинения для анализа русскоязычных текстов. Проведено исследование методов анализа текста, способов описания синтаксической структуры. Разработана система построения дерева синтаксического подчинения для анализа текстов русского языка, позволяющая выполнять обработку и анализ текстов больших объемов.

Annotation:

Lovkin A.V., Tarasova I.A. Development of a system for building a tree of syntactic subordination for the analysis of Russian texts. A study of text analysis methods, ways to describe the syntactic structure. A system for constructing a tree of syntactic analysis for the analysis of texts in Russian has been developed.

Постановка задачи

Автоматический анализ естественно-языковых текстов является востребованной технологией, которая находит применение в текстовых процессорах (например, Microsoft Word, OpenOffice.org Writer) и поисковых системах, системах реферирования, системах классификации и кластеризации текстов и, наконец, в системах поиска дубликатов в текстах.

Задача синтаксического анализа является одной из сложных задач компьютерной лингвистики. Исследования в этой области начались еще в 1960х годах. Были созданы различные системы, которые позволяли проводить синтаксический анализ предложений на естественном языке. Эти разработки существенно продвинули теорию и практику синтаксического анализа, однако, полученные программные реализации не обеспечивают высокое качество анализа.

Цель статьи – минимизация времени на обработку и анализ русскоязычных текстов больших объемов за счет разработки системы построения дерева синтаксического подчинения.

Для достижения поставленной цели были решены следующие задачи:

- проведено исследование методов анализа текста;
- проведено исследование способов описания синтаксической структуры;
- разработана система построения дерева синтаксического подчинения для анализа

текстов русского языка.

Исследования

Синтаксический анализ текста включает в себя следующие этапы:

- разбиение текста на базовые элементы;
- морфологический анализ;

- графематический анализ;
- построение дерева синтаксического подчинения.

Синтаксический анализ предназначен для выделения лексических и нелексических единиц текста в целях дальнейшей их обработки синтаксическим и семантическим компонентами лингвистического процессора [1,3].

Этап разбиения текста на базовые элементы предполагает выделение в тексте последовательностей символов одного алфавита (алфавит символов кириллицы, алфавит символов латиницы, алфавит знаков препинания, алфавит разделителей, алфавит для записи числовых значений, алфавит скобочек и кавычек, все остальные символы). Такие последовательности символов называются базовыми элементами.

На этапе морфологической разметки текста выполняется морфологический анализ написаний базовых элементов, состоящих из символов латиницы или кириллицы.

Этап графематического анализа позволяет интерпретировать базовые элементы и последовательности базовых элементов как нелексические единицы с определенной семантической нагрузкой (e-mail, URL, дата, имя файла и подобные им) или лексические единства, выступающие в предложении как единое целое [2].

Задачей синтаксического анализа является построение синтаксической структуры входного предложения (осуществление разбора предложения) на основе морфологической информации о словоформах и синтаксических правил объединения слов и словосочетаний. Синтаксическая структура отражает синтаксические связи, существующие между словами в предложении. Ее получение начинается с построения всевозможных связей между словами, которые в последующем отсеиваются на основе локальных и глобальных «фильтров». Конкретный вид структуры определяется выбранной системой синтаксических отношений (ССИО) [4,5].

Существует несколько способов описания синтаксической структуры, но два из них – система составляющих и дерево зависимостей – являются наиболее эффективными.

Для описания синтаксической структуры с помощью деревьев зависимостей (деревьев синтаксического подчинения) необходимо определить на множестве X точек цепочки x бинарное отношение таким образом, чтобы граф был (ориентированным) деревом с корнем. Всякое такое отношение называется отношением (синтаксического) подчинения, а соответствующее дерево – деревом (синтаксического) подчинения. При графическом изображении дерева подчинения, обычно располагают точки или, так называемые, цепочки на горизонтальной прямой и проводят стрелки сверху от нее. Тем самым получается «естественное» дерево подчинения, изображённое на рисунке 1.



Рисунок 1 – Дерево синтаксического подчинения (дерево зависимостей)

Корнем дерева подчинения принято считать сказуемое, т.к. оно представляет собой организующий элемент предложения.

Дуги дерева подчинения часто снабжаются метками, указывающими типы представляемых этими дугами синтаксических связей. Системы составляющих и деревья зависимостей характеризуют синтаксическую структуру предложения в разных аспектах. С помощью систем составляющих описываются в явном виде словосочетания, но игнорируется ориентация связей.

Одно предложение может допускать несколько различных "естественных" систем составляющих (деревьев подчинения). Чаще всего это бывает в тех случаях, когда смысл предложения можно понимать по-разному, и разные системы составляющих (деревья подчинения) отвечают разным толкованиям смысла [6].

Иногда для представления синтаксической структуры предложения используют смешанное представление, называемое обобщенной синтаксической структурой (ОСС). ОСС выражает, как и дерево зависимостей, ориентацию связей, но, в отличие от дерева зависимостей, ОСС снабжена информацией о словосочетаниях, образованных группами членов предложения (группой подлежащего, группой сказуемого, группой дополнений, группой обстоятельств и т.п.).

Реализация системы построения дерева синтаксического подчинения для анализа текстов русского языка осуществлена в Visual Studio 2010 в приложении MFC. Система позволяет анализировать заданный текст, передаваемый из файла, разбивать его на сегменты и выводить в виде дерева синтаксического подчинения.

На рисунке 2 представлен вид окна, которое выводится на экран при запуске. Для дальнейшей работы системы пользователю необходимо задать файл с текстом.

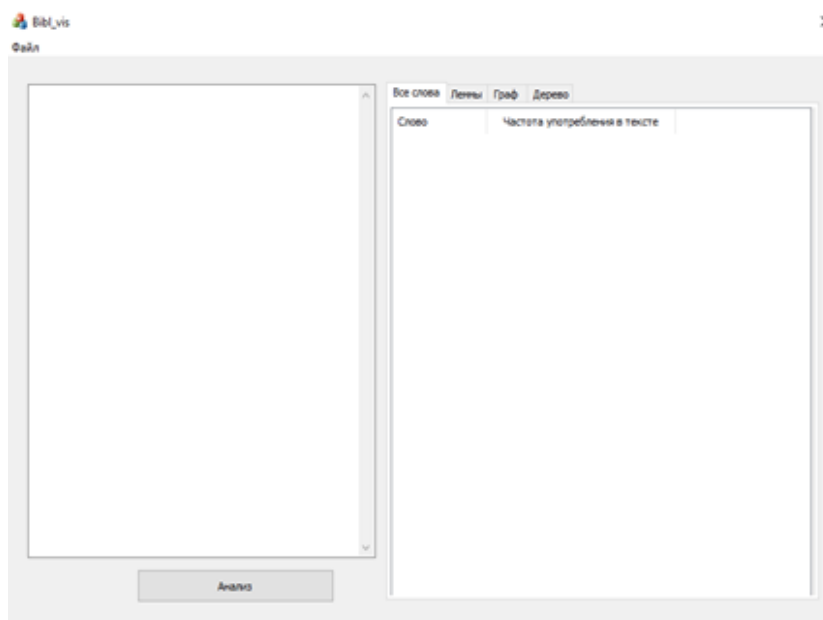


Рисунок 2 – Главное меню

На рисунке 3 представлен результат разбиения исходного текста на сегменты.

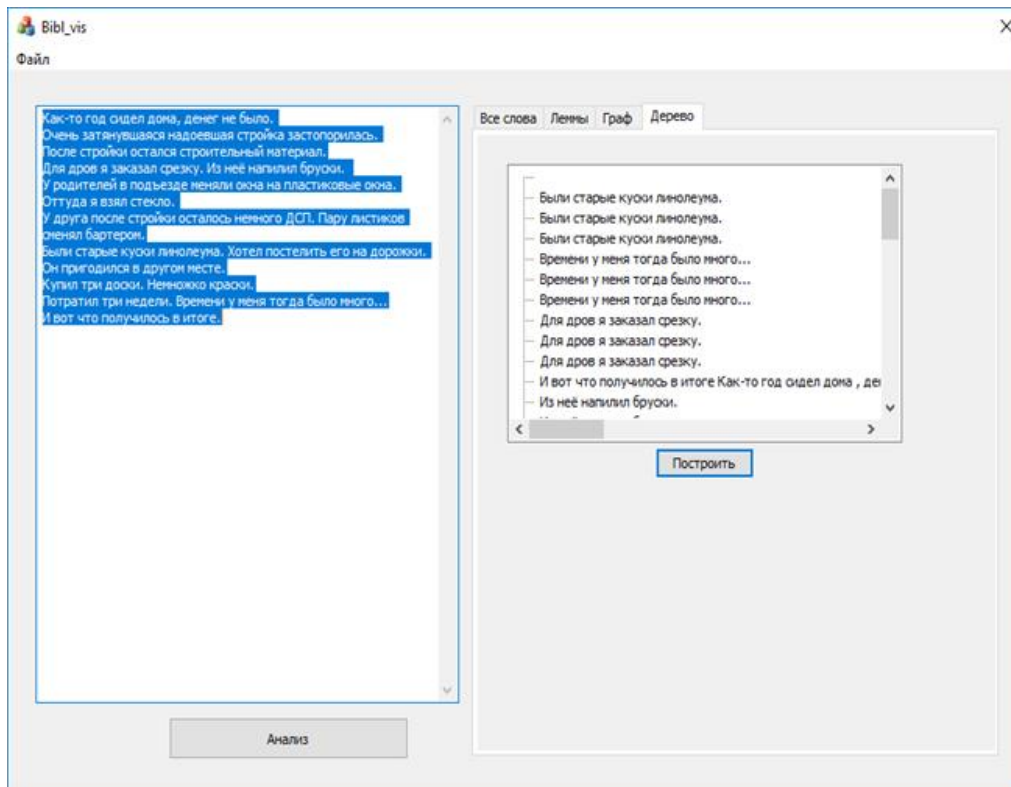


Рисунок 3 – Результат разбиения текста на сегменты

На рисунке 4 представлен результат описания синтаксической структуры с помощью дерева синтаксического подчинения.

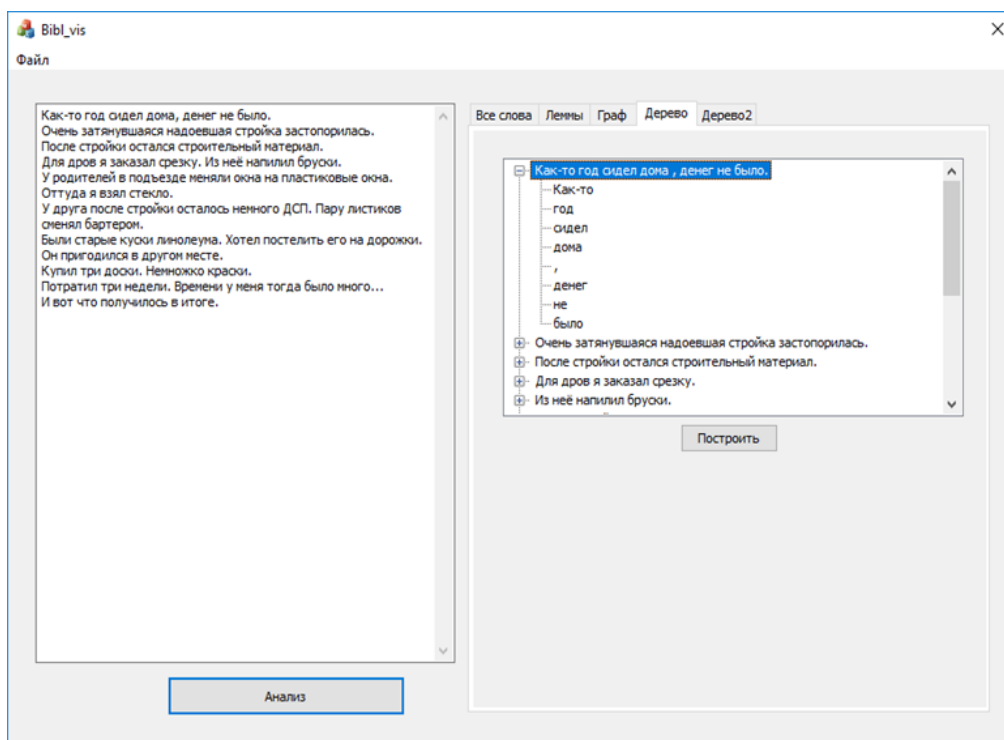


Рисунок 4 – Результат построения дерева синтаксического подчинения

Выводы

Разработанная система синтаксической обработки текста, описанная в данной работе, от существующих отличается тем, что морфологическая разметка выполняется до графематического анализа, после чего строится дерево синтаксического подчинения, позволяющее сохранить ориентацию связей. Это дает возможность дальнейшего использования дерева синтаксического подчинения в других задачах анализа текста.

Литература

1. Автоматическая Обработка Текста [Электронный ресурс].– Режим доступа: <http://www.aot.ru/technology.html>
2. Сокирко, А.В. Семантические словари в автоматической обработке текста / А.В. Сокирко // Диссертация на соискание ученой степени кандидата технических наук. – М.: МГПИИЯ. 2001. – 108 с. [Электронный ресурс]. – Режим доступа – <http://www.aot.ru/docs/graphan.html>
3. Дорохина, Г.В. Модуль морфологического анализа слов русского языка / А.П. Павлюкова, Г.В. Дорохина // Искусственный интеллект. Донецк: – ИПШ: Наука і освіта, 2004. – № 3. – С. 636-642.
4. Дорохина, Г.В. Модуль морфологического анализа без словаря слов русского языка / Г.В. Дорохина, В.Ю. Трунов, Е.В. Шилова // Искусственный интеллект. – №2. – 2010. – С.32-36.
5. Бондаренко Е. А. Принципы автоматической обработки естественно-языковых текстов: валентностный подход / Е.А. Бондаренко, О.А. Каплина // Искусственный интеллект. – 2013. – №1. – С. 80-90.
6. Galina V. Dorokhina The Algorithm of Syntactic Analysis Based on Grammatical Rules // Искусственный интеллект. - №4, 2014. – С. 169 – 179.

Научное издание

**ИНФОРМАТИКА, УПРАВЛЯЮЩИЕ СИСТЕМЫ,
МАТЕМАТИЧЕСКОЕ И КОМПЬЮТЕРНОЕ
МОДЕЛИРОВАНИЕ
(ИУСМКМ-2019)
(Студенческая секция)**

Материалы студенческой секции X Международной
научно-технической конференции в рамках
V Международного Научного форума
Донецкой Народной Республики

22-24 мая 2019 г.

Редактор: Р. В. Мальчева

Дизайн и верстка: М. П. Руденко

Web-сайт конференции: <http://www.iuskm.donntu.org>