

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО
ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ
РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ В. Ф. УТКИНА

**ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ**

Межвузовский сборник научных трудов

Рязань 2021

УДК 004

Информационные технологии, межвузовский сборник научных трудов. – Рязань: ИП Коняхин А.В. (Book Jet), 2021. – 207 с.

ISBN ; 9: /7/; 29622/67/2

Публикуются статьи о проблемах использования информационных технологий в науке и образовании.

Сборник рассчитан на научно-педагогических работников вузов и может быть использован студентами и аспирантами инфокоммуникационных и информационных специальностей.

Редакционная коллегия:

д-р техн. наук, проф. В.П. Корячко (ответственный редактор), д-р техн. наук, проф. Ю.Л. Леохин (Московский технический университет связи и информатики), д-р техн. наук, проф. В.П. Кулагин (МИРЭА – Российский технологический университет), д-р техн. наук, проф. А.Д. Иванников (Институт проблем проектирования в микроэлектронике Российской академии наук), д-р техн. наук, проф. С.В. Скворцов (РГРТУ), д-р техн. наук, проф. Д.А. Перепелкин (РГРТУ), канд. техн. наук А.Н. Сапрыкин (ответственный секретарь).

Рецензенты:

д-р техн. наук, проф. А.И. Таганов (Рязанский государственный радиотехнический университет имени В.Ф. Уткина), д-р техн. наук, проф. В.А. Минаев (Московский государственный технический университета имени Н.Э. Баумана).

ISBN ; 9: /7/; 29622/67/2

©Рязанский государственный
радиотехнический университет
имени В.Ф. Уткина, 2021
©ИП Коняхин А.В. (Book Jet), 2021

СОДЕРЖАНИЕ

Андреев Д.А., Силкин А.В. Применение модели нейронных сетей в задачах радиоразведки.....	8
Брыгалин А.О. Самые значимые достижения второго десятилетия 21 века в разработке аккумуляторов.....	12
Брыгалин А.О. Четвертая промышленная революция	18
Васюкова Ю.А., Газаринская М.С. Применение технологии лазерной селективной пайки при монтаже печатных узлов.....	22
Давыдкин К.А. Модель нагрузка-прочность при ускоренных испытаниях на надежность	25
Данилов А.Д., Сускин В.В. Структура и принцип работы системы охлаждения на основе термоэлектрических модулей	30
Дураев И.В. Анализ современных приемопередающих интерфейсов.....	35
Дураев И.В. Исследование возможностей применения подогрева РЭА	38
Жильцов А.В. Переключатель состояния на основе модели Т-триггера на ПЛИС.....	41
Жукова О.А. Применение программных средств, реализующих алгоритм экспертного оценивания на основе медианы Кемени	47
Захаров И.С. Настраиваемый беспилотный наземный автомобиль с дистанционным управлением	50

Зинова В.О., Черенкова А.Б.

Разработка модифицированного генетического алгоритма для компоновки блоков электронно-вычислительных средств и его применение для адаптации данных 53

Иванчикова М.А., Турбина Д.С.

Анализ данных для прогнозирования потока абитуриентов по направлениям подготовки высшего образования 57

Игошина М.С., Яшкова В.В.

Сравнительный анализ алгоритмов размещения 60

Калинин Т.А., Панкова М.А.

Сравнение алгоритмов построения деревьев Штейнера в ортогональной метрике 65

Кащеев А.Н.

Вычисление математических функций на языке C++ 71

Киреев П.С., Скворцов С.В.

Применение свёрточных нейронных сетей для решения задачи распознавания изображений 74

Кожиков В.Ю., Сенькина Е.А.

Многофункциональные панели операторов 79

Кочергин Э.Г., Трубицын А.А.

Движение параксиальных пучков электронов в аксиально-симметричном поле 85

Кошелев А.Д.

Генетические алгоритмы в вычислительной технике 92

Крыгина М.К., Чусова А.А.

Визуализация сейсмоакустической информации 98

Лебедева Н.С., Сапрыкин А.Н.

Поиск оптимальных базовых параметров генетического алгоритма для балансировки сетевых потоков в ЦОД 104

Лебедева Н.С., Сапрыкин А.Н.

Применение генетического алгоритма для балансировки сетевых потоков в ЦОД 110

Ликучев В.Ю., Перепёлкин Д.А.

О многоагентном подходе к решению задач размещения радиоэлектронных компонентов 115

Маркова Н.А.

Обзор и исследование микроконтроллеров 122

Маслова А.Д., Плаксина А.Р.

Концептуальные основы мобильного обучения студентов 126

Музюкин И.А., Сапрыкин А.Н.

Модифицированный оператор миграции в островной модели генетического алгоритма для компоновки блоков ЭС 129

Новиков Я.И.

Интерполирование функций и построение интерполяционных многочленов 132

Осин И.Г.

Критерии выбора конфигурации и сравнительный анализ искусственных нейронных сетей 137

Осин С.В.

Механизм аутентификации указателей в процессорах на архитектуре ARM 141

Панинская Д.С., Сускин В.В.

Устройство стыковки и перемещения чеки специального назначения 148

Поверенный Г.В.

Использование ветроустановки в системе электроснабжения 152

Поверенный Г.В.

Целесообразность внедрения ветроэнергетических установок в условиях России 156

Потапкина Н.С.

Конструирование и проектирование вычислительного комплекса передачи и обработки данных в программно-конфигурируемых сетях 160

Приходько Р.А.

О переходе к микросервисной архитектуре программного обеспечения 163

Размахнин И.Д.

Разработка программных средств компоновки элементов с использованием матричного алгоритма 168

Савилов И.В., Стручков П.Е.

Альтернативные источники энергии в частном доме 171

Салынов П.А., Скворцов С.В.

Применение рекуррентных нейронных сетей для решения задачи распознавания речи 177

Силкин А.В.

Применение радиоустройств с программируемыми параметрами в радиотехнических системах 181

Стеблюк А.Ю.

Особенности организации и функционирования информационно-аналитических систем 185

Ушанов А.С.

Стереозрение в задачах обнаружения и траекторного сопровождения объектов 190

Филькин К.Ф.

Обеспечение электромагнитной совместимости корректора коэффициента мощности вторичного источника электропитания 193

Филькин К.Ф.

Программный подход реализации алгоритма управления ККМ на основе микроконтроллеров 196

Шевелёв В. А.

Принцип работы фильтра нижних частот 199

Юрков Н.Ю.

Чатбот в Telegram как основа плагина для LMS Moodle..... 203

УДК 004.032.26

Д.А. АНДРЕЕВ, А.В. СИЛКИН

Межвидовой центр подготовки и боевого применения войск РЭБ
(учебный и испытательный)

ПРИМЕНЕНИЕ МОДЕЛИ НЕЙРОННЫХ СЕТЕЙ В ЗАДАЧАХ РАДИОРАЗВЕДКИ

Рассматривается возможность применения модели нейронных сетей для классификации перехватываемых в ходе радиоразведки речевых сообщений.

Одной из задач радиоразведки является классификация по языковой принадлежности перехватываемых речевых сообщений от обнаруженных в ходе радиомониторинга источников радиоизлучения. Прослушивание речевых сообщений оператором средства радиоразведки и ручная классификация занимает длительное время. Зачастую объем перехватываемой информации превышает рабочие возможности оператора. Поэтому актуальной является разработка программных модулей автоматической классификации сообщений.

Рассмотрим работу модулей, в основе которых лежит модель сверточной нейронной сети (CNN), реализованной на языке Python.

Нейронные сети CNN схожи с обычными нейронными сетями. Они состоят из нейронов с обучаемыми весами и сдвигами. Каждый нейрон получает входные данные, выполняет скалярное произведение и добавляет нелинейную оптимизацию при необходимости. Вся сеть выражает одну дифференцируемую функцию оценки. Она способна обнаруживать первичные функции, которые затем объединяются последующими слоями архитектуры CNN, что приводит к обнаружению более сложных и релевантных признаков [1].

Классификация звуковых файлов, содержащих речевые сообщения, с помощью CNN на основе библиотеки *Keras* языка Python осуществляется двумя способами. Первый заключается в непосредственной обработке звукового файла и включает в себя извлечение акустических признаков, относящихся к поставленной задаче. Второй способ, как и первый, включает в себя извлечение акустических признаков, но формирует данные для обучения из скопированных в качестве изображений спектрограмм.

Признаки – основа обучения нейронной сети. Каждое речевое сообщение (рисунок 1) состоит из множества признаков, таких как частотные компоненты, частота основного тона, спектральная плотность, спектральный центроид, спектральный спад и т.д.

Спектральный центроид, например, указывает на какой частоте сосредоточена энергия спектра или, другими словами, указывает на расположение «центра масс» для звука (рисунок 2). Спектральный спад является мерой формы сигнала, представляющей собой частоту, в которой высокие частоты снижаются до нуля (рисунок 3).

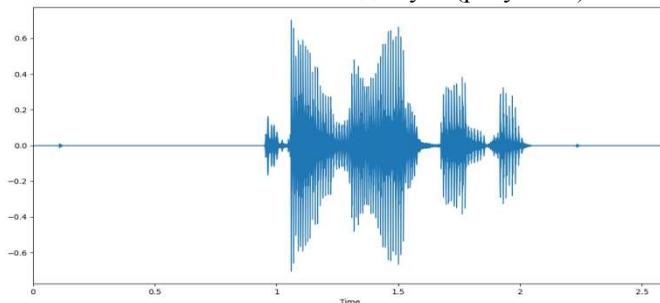


Рисунок 1 – Речевое сообщение «Ботаника – наука о растениях»

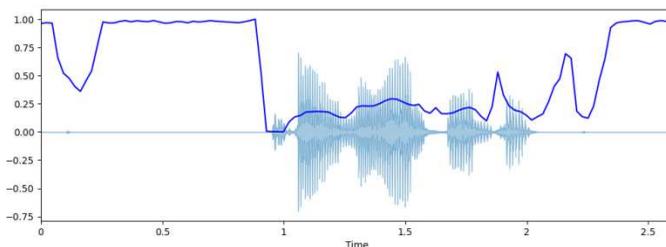


Рисунок 2 – Извлечение спектрального центроида

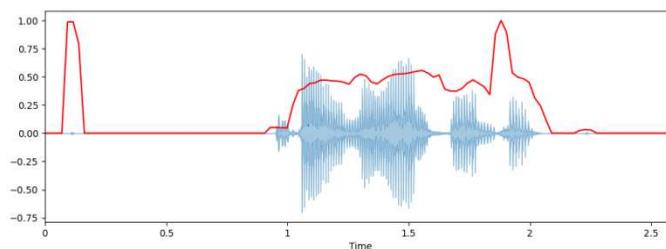


Рисунок 3 – Извлечение спектрального спада

Извлечение большинства из этих признаков помогает точнее определять языковую принадлежность речевого сообщения.

Одним из ключевых признаков являются мел-кепстральные коэффициенты, полученные в результате применения мел-шкалы к результатам быстрого преобразования Фурье исходных данных и

моделирующие характеристики человеческого голоса (рисунок 4). Особенности применения мел-кепстрального анализа заключаются в следующем [2]:

- спектр проецируется на специальную мел-шкалу, позволяя выделить наиболее значимые для восприятия человеком частоты;
- количество вычисляемых коэффициентов может быть ограничено любым значением, что позволяет варьировать количеством обрабатываемой информации.

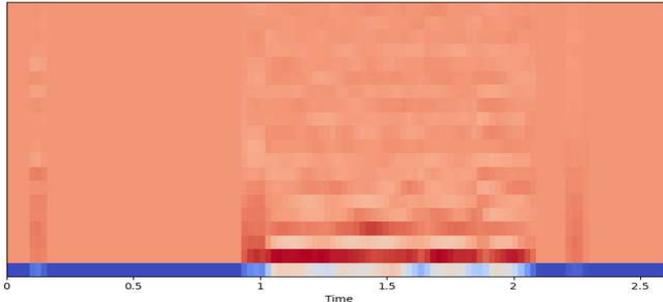


Рисунок 4 – Мел-частотный спектр речевого сообщения

Рассмотрим реализацию первого способа. Основная библиотека расчета акустических признаков – *Librosa*, позволяющая также отобразить графики массивов аудиофайлов и спектрограмм. Результатом работы библиотеки является файл с расширением *.csv*, содержащий набор акустических признаков. Библиотека *StandardScaler* позволяет сформировать из файла вектор данных для непосредственной работы с моделью нейронной сети, реализованной *Keras*.

Данный способ хорошо зарекомендовал себя при работе с речевыми сообщениями, состоящими из нескольких слов. При увеличении длительности фраз оптимальным вариантом является переход ко второму способу.

Реализация второго способа основана на применении многослойной модели сверточной нейронной сети, для обучения которой применяют спектрограммы речевых файлов (рисунок 5). Данный способ позволяет перейти от обработки массивов данных к обработке изображения. Первым этапом является создание с помощью библиотеки *Librosa* спектрограмм и созданием директории, содержащей все изображения, классифицированные по языкам. Следующий шаг – развитие данных на наборы для обучения и тестирования нейронной сети с помощью библиотеки *SplitFolders* и непосредственная работа с моделью нейронной сети, реализованной

Keras. Для увеличения набора обучающих данных возможно использовать искусственное создание обучающих изображений с помощью различных способов обработки, таких как случайная ротация, переходы, сдвиги и перевороты [3].

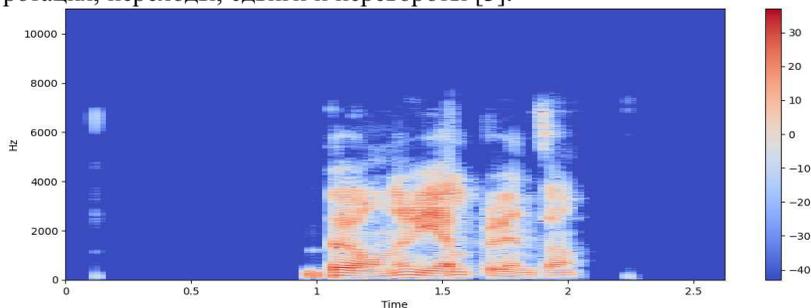


Рисунок 5 – Спектрограмма речевого сообщения

Подобная классификация речевых сообщений по языковой принадлежности с применением модели нейронных сетей так же применима и к идентификации отдельных абонентов. Для более точного определения абонента потребуется большее количество обучающих речевых сообщений, но точность при этом останется на высоком уровне.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Ballard W. Hands-On Deep Learning for Images with TensorFlow. UK: Packt Publishing. 2018.
2. Жерон О. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow. Концепции, инструменты и техники для создания интеллектуальных систем. СПб: Альфа-книга. 2018.
3. Ketkar N. Deep Learning with Python: A Hands-on Introduction. Apress. 2017.

УДК 621.355.9

А.О. БРЫГАЛИНРязанский государственный радиотехнический университет
имени В.Ф. Уткина**САМЫЕ ЗНАЧИМЫЕ ДОСТИЖЕНИЯ ВТОРОГО
ДЕСЯТИЛЕТИЯ 21 ВЕКА В РАЗРАБОТКЕ АККУМУЛЯТОРОВ**

В статье рассматриваются технологии, по которым делаются АКБ в последнее десятилетие. Описываются плюсы и минусы данных технологий.

Разработка аккумуляторов - очень перспективное направление научных исследований, в котором постоянно происходят открытия. 2020 год показал, насколько изобретательными могут быть ученые, когда дело доходит до разработки новых технологий хранения электроэнергии.

Уже в скором времени традиционные литий-ионные батареи будут не актуальны с их текущими характеристиками. В прошедшем году ученые предлагали самые разные варианты улучшить аккумуляторы, сделав их мощнее, экономичнее и эффективнее.

Батареи на безопасном металлическом литии.

Один из самых перспективных новых материалов — металлический литий. Его использование в качестве анода, может серьезно повысить плотность аккумулятора и продлить срок его службы. Но металлический литий опасен: при зарядке на аноде создаются наросты (дендриты), что ведет к коротким замыканиям и возгоранию — не очень заманчивая перспектива.

Ученые из университета штата Вашингтон предложили добавить несколько химикатов в раствор катода и электролита. В итоге на поверхности анода образовывался защитный слой, и анод мог оставаться стабильным при зарядке в течение 500 циклов. Преимущество технологии - возможность интегрировать ее в существующие производственные процессы.

Аккумулятор с ультразвуком устройством.

Ученые из Калифорнийского университета в Сан-Диего очень творчески подошли к решению проблемы металлического лития. Они разработали миниатюрное ультразвуковое устройство и включили его в литий-металлическую батарею. Устройство посылает высокочастотные звуковые волны через жидкий электролит, заставляя его течь плавно, а не находиться статично.

Это, в свою очередь, приводит к равномерному распределению лития на аноде, и дендриты не накапливаются. При тестировании аккумулятора с новым оборудованием заряжался от 0 до 100% всего за 10 минут и продержался 250 циклов зарядки.

Самые быстрые электроды.

Специалисты французского Nawa Technologies решили изменить структуру электродов в обычных литий-ионных аккумуляторах - их мощность значительно увеличилась. Вместо беспорядочной структуры, требующей, чтобы заряженные ионы перемещались через лабиринт, разработчики предложили электрод, который состоит из выровненной по вертикали структуры, похожей на щетку для волос, с высокопроводящими углеродными нанотрубками.

Это создает быструю полосу для движущихся ионов и позволяет батарее заряжаться в 10 раз быстрее, заряжая батарею от 0 до 80% всего за 5 минут. Срок службы увеличен в 5 раз, а удельная энергия - в 3 раза.

Батарея с использованием кремниевого анода.

Кремний может хранить в 4 раза больше ионов лития, чем современные графит и медь, но ёмкость батарей быстро уменьшается. Ученые из Кореи нашли способ избежать этого. Кремниевый анод помещается в специальный раствор, который заставляет электроны и ионы лития просачиваться в электрод, таким образом, компенсируются потери

Обычно аноды на основе кремния теряют более 20% ионов лития во время зарядки, а новый анод теряет менее 1%. Кроме того, его удельная энергия на 25% выше, чем у имеющихся в продаже аналогов.

Нанонити для увеличения емкости.

Российские ученые из МИЭТ разработали нанонити, которые способны в 3 раза увеличивать ёмкость аккумуляторов, и благодаря им тепловые потери при работе могут быть преобразованы в электричество.

По мнению ученых, германиевые нанонити могут заменить графитовые электроды в современных батареях. При этом их объём увеличивается, но габариты не изменяются[2].

Натрий-ионные аккумуляторы из пластиковых бутылок.

Еще один вид аккумуляторов с большим потенциалом — это натрий-ионные. Ученые из университета Пердью взяли за основу обычные пластиковые отходы для создания ключевого компонента такой батареи.

Для превращения пластика в хлопья они применили стандартную микроволновую технологию, обработав его сверхбыстрым микроволновым излучением. В результате ученые получили терефталат натрия, известный своими хорошими электрохимическими характеристиками и пригодный для производства натрий-ионной батареи.

Новое покрытие для старой батареи.

В Аргоннской национальной лаборатории нашли способ продлить срок службы привычных литий-ионных аккумуляторов — для этого использовали инновационное катодное покрытие из полимера PEDOT (поли-3,4-этилендиокситиофен).

Благодаря такому покрытию традиционные аккумуляторы становятся безопаснее, так как не образуется нежелательная пленка на катоде. При этом, срок их службы продлевается, так как повышается рабочее напряжение.

Электроды из панцирей креветок.

Хитин, содержащийся в панцирях креветок, в Массачусетском технологического институте использовали для производства экологически безопасного электрода для проточной батареи. Вместо того, чтобы хранить энергию внутри самого аккумулятора, такие батареи сохраняют энергию в жидких электролитах в больших внешних резервуарах, которые при необходимости можно просто увеличить.

Ученые решили сделать главный строительный блок этих батарей из экологически чистых материалов. Для изготовления электродов проточной батареи хитин брали из панцирей креветок в сочетании с войлоком. Помимо высокой удельной мощности такой аккумулятор имеет еще одно преимущество - невысокую стоимость исходного материала[2].

Самый прочный электролит в мире.

Еще один пример твердотельной батареи с впечатляющей долговечностью — изобретение исследователей из Университета Брауна. Они утверждают, что, добавив графен в керамический материал, сделали самый прочный твердый электролит на свете.

Любопытно, что графен обладает высокой проводимостью, что просто нежелательно для электролита аккумулятора. Но, удерживая концентрацию графена на довольно низком уровне, ученые смогли найти золотую середину, которая не позволяла графену проводить электричество, но в то же время обладала высочайшей прочностью.

Жидкометаллическая батарея, работающая при комнатной температуре.

Как правило, в жидкометаллических батареях металлы должны быть нагреты до температуры не менее 240°C . Но экспериментальная полностью жидкометаллическая батарея способна работать при комнатной температуре и имеет гораздо лучшие характеристики, чем литий-ионные батареи.

В нем ученые из Техасского университета в Остине использовали сплавы, которые могут оставаться жидкими при комнатной температуре. Такой аккумулятор имеет большую емкость, чем литий-ионный аккумулятор, и заряжается намного быстрее.

Неупорядоченность на страже электрического заряда.

Ученые Иллинойского университета в Чикаго разработали новую технологию производства аккумуляторных батарей для мобильных устройств, в основе которой лежит принцип использования неупорядоченных частиц оксида магния и непосредственно магниевых анода. Эту идею до них никто не развивал, поскольку неупорядоченные (или беспорядочные, движущиеся хаотично) частицы теоретически и практически могут стать препятствием при производстве и эксплуатации элементов питания.

Несмотря на то, ранее технология нигде и никем не применялась, американские ученые уже добились определенных успехов в выбранном направлении. К главным преимуществам магниевых АКБ авторы технологии отнесли их повышенную безопасность в сравнении с литиевыми батареями, а также способность гораздо дольше сохранять заряд.

По словам исследователей, если литий-ионные аккумуляторы уже достаточно давно достигли пика своего развития, то магниевые лишь только начинают свой путь, имея в запасе огромный потенциал.

Суть и возможность технологии.

В аккумуляторе, созданном учеными Иллинойского университета, используется созданный ими на основе оксида магния и хрома (MgCr_2O_4) неупорядоченный материал толщиной порядка 5 нанометров. Его характеризует в первую очередь низкая температура реакции при высокой скорости этой самой реакции. На практике это даст возможность не опасаться перегрева аккумулятора в мобильном устройстве в жаркий летний день или в процессе подзарядки. Литий-ионные батареи, отметим, очень чувствительны к изменению температуры и могут воспламениться и даже взорваться прямо в руках у владельца смартфона.

Преследуя цель убедиться в своей правоте, ученые провели сравнительный эксперимент, в ходе которого сопоставили 5-нанометровый неупорядоченный материал с 7-нанометровым упорядоченным оксидом магния и хрома. Оба материала подвергались различным испытаниям и тестам, включая рентгеновскую абсорбционную спектроскопию и современные электрохимические методы тестирования.

Эксперты исследовали структурные и химические изменения материалов во время испытаний и увидели, что они ведут себя совершенно по-разному. Неупорядоченные частицы оксида магния могут течь от анода к катоду, а упорядоченные частицы - нет. На основании полученных результатов ученые пришли к выводу, что их новая технология подходит для создания аккумуляторных батарей нового типа. По состоянию на декабрь 2018 года технология требовала доработки и не могла быть использована в массовом производстве.

Магний лучше лития, никеля и кадмия?

О применении магния в перезаряжаемых элементах питания специалисты стали задумываться еще в начале века, даже когда литиевые батареи еще не получили столь широкого распространения. В 2003 г. израильские ученые из университета в Рамат-Гане даже разработали прототип нового магниевый аккумулятора, который практически не уступал по своим энергетическим свойствам популярным тогда никель-кадмиевым АКБ. Он тоже выдавал напряжение до 1,2 В, но при этом характеризовался меньшей степенью деградации спустя несколько сотен циклов зарядки и разрядки и в целом был намного более экологичным. В серию аккумуляторы, выполненные по израильской технологии, не пошли[3].

Конкурирующие разработки.

Есть и другие технологии, которые могут заменить литиевые батареи и положить конец их далеко не экологически чистому производству и утилизации. Например, другая группа американских ученых, на этот раз из Калифорнийского технологического института, создала батарею на основе фторидов - химических соединений фтора с другими элементами периодической таблицы. Такие АКБ теоретически характеризуются способностью удерживать заряд до восьми раз дольше, чем литий-ионные и литий-полимерные. Опять же, они намного безопаснее, чем другие, потому что они не подвергаются воздействию повышенных температур окружающей среды или тепла во время подзарядки.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Дискретная математика: Учеб. пособие / А. М. Гостин, В. П. Корячко, М. А. Бакулева, А. В. Бакулев - РГРТУ, 2011. - 135 с.
2. Значимые достижения 2020 года [Электронный ресурс]. – URL:https://zoom.cnews.ru/rnd/article/item/samy_e_znachimye_dostizheniya_2020_goda_v_razrabotke_akkumulyatorov (дата обращения 5.04.2021).
3. Создан аккумулятор будущего [Электронный ресурс]. – URL:https://www.cnews.ru/news/top/2018-1219_uchenye_sozdali_nadezhnyj_i_dolgoigrayushchij_akkumulyator (дата обращения 5.04.2021).

УДК 519.256

А.О. БРЫГАЛИН

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

ЧЕТВЕРТАЯ ПРОМЫШЛЕННАЯ РЕВОЛЮЦИЯ

В статье описаны основные моменты, которые необходимо учитывать при создании сетей хранения данных.

Четвертая промышленная революция, которую часто называют «переходом к индустрии 4.0», находится в самом разгаре, порождая несколько интересных трендов. Один из них связан с тем, что компании увеличивают объем своих инвестиций и выделяют все большее количество ресурсов на сбор и хранение данных, получаемых из сетей интернета вещей (IoT) и оборудования промышленного интернета вещей (IIoT) в составе сборочных линий и ИСУП. А также от контролируемых операций, которые объединяют их сервера по всему миру.

Постепенно внедряя 5G, искусственный интеллект (ИИ) и машинное обучение (МО) – перспективные технологии, чей расцвет ожидается в ближайшие годы, компании промышленного сектора должны задаться вопросом, как они смогут обрабатывать и хранить огромные объемы данных, которые они, вероятно, будут генерировать.

Развитие операционной деятельности компаний.

Индустрия 4.0 это наглядно показывает постоянную эволюцию подхода к ведению бизнеса. Сегодня, учитывая огромное число подключенных устройств, для выполнения рутинных операций компании гораздо сильнее полагаются на автоматизацию и на сбор и ведение журналов данных. Дальнейшее технологическое развитие, в частности, внедрение 5G, только подстегнет этот тренд, обеспечивая необходимые для работы подключенных устройств скорость и мощность за счет возможностей подключения и взаимодействия следующего поколения.

В условиях такого увеличения объема данных, требующих обработки, вычислительные мощности будут все чаще сдвигаться ближе к конечным устройствам и локальным ЦОД, чтобы как можно больше снизить нагрузку на центральные ЦОД.

Итак, в условиях четвертой промышленной революции, проходящей под знаменем данных, насколько обоснованно говорить о

том, что хранение данных – это ключ к успеху промышленного интернета вещей (IIoT)[2]?

1. Внешние факторы.

Существует заблуждение о том, что СХД должна и будет работать одинаково, независимо от места расположения, но, говоря коротко, это совсем не так. Окружающие условия влияют на все, в том числе и на хранение данных, поэтому не брать их в расчёт нельзя. При выборе оборудования для хранения данных компании для каждой конкретной ситуации должны учитывать целый ряд внешних факторов, в том числе высоту над уровнем моря, влажность, температуру воздуха и вибрационные нагрузки (перечень не является исчерпывающим).

Например, умный автомобиль, работающий в жаркой пустыне или на севере, должен быть надлежащим образом оборудован, чтобы выдерживать экстремальные погодные условия, а приводы, установленные на удаленных морских буровых установках, должны быть устойчивыми к вибрации и, возможно, к влаге и сырости.

2. Производительность.

Требования, которые предъявляются к вычислительным мощностям и общей производительности СХД могут очень сильно варьироваться. Производительность, которой обладало старое устройство, едва ли соответствует требованиям сегодняшнего дня. Представьте хотя бы запросы со стороны промышленных дронов с элементами ИИ, и разница станет очевидной. К используемым в таких условиях устройствам, обеспечивающим хранение и обработку огромных объемов сгенерированных ИИ данных, предъявляются очень высокие требования, и об этом следует помнить в контексте вечно «голодной» до данных технологии IIoT.

3. Износостойкость.

Приложения с интенсивным чтением и записью в значительной степени полагаются на надежность и отказоустойчивость решения хранения, которое должно иметь возможность записывать данные в течение длительного периода времени. Это соображение становится особенно важным при установке оборудования (например, камер теплогрaфии) в труднодоступных местах.

В таких случаях все устройства должны быть надежными, чтобы для их установки и бесперебойного функционирования требовалось лишь ограниченное техническое обслуживание. Хотя производительность – это важная характеристика для устройств хранения данных, при определенных вариантах использования скорость обработки данных не так важна, как общая надежность

оборудования, а значит, если того требует конкретная задача, производителям придется пожертвовать первым ради второго.

4. Дистанционный мониторинг.

Когда дело доходит до мониторинга данных, возникает ряд вопросов. Как осуществляется этот мониторинг? У кого есть доступ к этим данным и как часто? На фоне недавнего развития локальных центров обработки данных физический доступ и обслуживание центров обработки данных и IoT - устройств не всегда кажутся практически оправданными. В этих условиях удаленный мониторинг состояния запоминающих устройств становится важным для периодического обслуживания и предотвращения ненужных проблем.

5. Длительное хранение данных.

Хранение данных в течение длительного времени зачастую件но для внутренней отчетности компаний как средство для принятия стратегических решений, однако, в некоторых случаях возможность такого хранения становится насущной необходимостью. Если компания обрабатывает и ведет журналы критических важных данных, будь то информация, связанная с госуправлением, или, например, медицинские записи, такие данные должны храниться в течение длительного периода времени как для того, чтобы соответствовать нормативным требованиям, так и для снижения числа возможных ошибок. В таких ситуациях возможность долгого хранения данных – это необходимость, требующая выбора правильного оборудования.

Информационным потребностям, связанным с четвертой промышленной революцией, редко уделяют первостепенное значение при создании и внедрении инновационных решений, хотя, вполне возможно, зря. При покупке новой технологии компаниям промышленного сектора следует начать с анализа уже имеющихся материально-технической базы, чтобы убедиться в достаточности мощностей их СХД для обеспечения эффективной работы новых высокотехнологичных устройств. Компаниям придется инвестировать в решения, которые удовлетворяют их потребностям, будь то накопители на флеше или жесткие диски (HDD-накопители)[2].

Интеллектуальные технологии увеличивают эффективность и производительность компаний за счет обработки и анализа данных с невиданной ранее скоростью. Индустрия 4.0 - это отражение того, как технологии меняют способ работы компаний, но весь потенциал технологических решений можно полностью раскрыть только с учетом требований к хранению.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Edward Rosten, Reid Porter and Tom Drummond: "Faster and better: a machine learning approach to corner detection", Department of Engineering, University of Cambridge, UK, 2008.

2. Четвертая промышленная революция [Электронный ресурс]. – URL: <https://www.it-world.ru/tech/technology/168810.html> (дата обращения 3.04.2021)

УДК 004.65

Ю.А. ВАСЮКОВА, М.С. ГАГАРИНСКАЯ

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

ПРИМЕНЕНИЕ ТЕХНОЛОГИИ ЛАЗЕРНОЙ СЕЛЕКТИВНОЙ ПАЙКИ ПРИ МОНТАЖЕ ПЕЧАТНЫХ УЗЛОВ

Данная статья посвящена обзору технологии и созданию программы пайки печатных плат на устройстве лазерной селективной пайке FireFly.

Система **Firefly** представляет собой установку для лазерной селективной пайки, она предназначена для присоединения электронных компонентов – резисторов, конденсаторов, интегральных схем – к печатным платам.

Соединение, получаемое в результате пайки, называемое пайкой с оловянным или твердым припоем, представляет собой соединение двух металлических поверхностей с использованием материалов с низкой точкой плавления. Припаяваемые части должны находиться на небольшом расстоянии друг от друга, поверхности смачиваются расплавленным припоем, затем припой должен охладиться до перехода в твердое состояние. Процесс пайки с твердым припоем предполагает использование только присадочных металлов, соединение с основным компонентом соединения обеспечивается благодаря явлению «смачивания» и капиллярной инфильтрации. Процесс происходит следующим образом: основной компонент соединения подготовлен так, чтобы между двумя соединяемыми деталями оставалось небольшое пространство, затем он нагревается до температуры более низкой, чем его собственная температура плавления, но выше температуры плавления присадочного металла. Это достигается при контакте основного компонента или источника тепла и затем, при выполнении определенных условий, происходит инфильтрация жидкой фазы в свободное пространство между соединяемыми деталями с одновременным смачиванием. По мере охлаждения присадочный металл затвердевает и если смачивание прошло успешно, то он соединяется с основным компонентом.

Технология лазерной селективной пайки обладает следующими преимуществами:

- пайка, которая не требует нагрева всего изделия;
- пайка в течение долей секунд;
- пайка без необходимости создания специальной атмосферы.

Внедрение новой технологии позволяет: уменьшить расход материалов; исключить операции раскатки припоя и нагрева изделия, что позволяет улучшить механические характеристики секции. Также данная технология позволяет сократить время на выполнение конструкционной пайки. За счет этого сокращается количество рабочих мест. Данные преимущества позволяют снизить себестоимость изделия.

Вышеперечисленные преимущества разрабатываемой технологии доказывают целесообразность ее разработки по экономическим, техническим и эксплуатационным показателям.

Выполнение программы пайки осуществляется с помощью ПО VIVA установленном на главном компьютере и управляющая работой системы Firefly.

Описание плат в ПО VIVA осуществлено с использованием информации, содержащейся в файлах CAD/CAM, которые применяются при изготовлении плат. ПО VIVA позволяет создавать эту информацию и импортировать данные файлов, поступающих из среды CAD/CAM.

Плата представлена в виде набора узлов. Атрибуты, отнесенные каждому узлу, позволяют определить узел в координатах платы XY; геометрические и технологические характеристики узла, вид операции для данного узла, некоторые другие атрибуты, которые не являются обязательными для процесса пайки, например, имя сигнала на узле.

Выполнение программы пайки для платы включает в себя выполнение паек, содержащих ссылки макрокоманды SOLD для данной платы.

При создании программы ПО VIVA для каждого узла, содержащего значения для атрибутов "SOLDERABILITY" создает ссылку макрокоманды SOLD с фактическими параметрами, рассчитанными соответственно атрибутам, которые определяют физические характеристики узла.

Создание программы пайки начинается с открытия диалогового окна на странице «Создать программу» "Create Program" вводится информация для создания программы пайки: В полях «Выбор припоя» "Alloy Selection", «Диаметр проволоки» "Wire Diameter" и «Флюс (Процент %)" "Flux (Percent %)" устанавливаются значения, соответствующие виду используемой проволоки припоя. В поле «Поверхность платы» "PCB Surface" выбирается вид поверхности дорожек на плате. В поле «Цвет платы» "PCB Color" указывается цвет платы. В разделе «Температура установки предварительного нагрева»

"Pre-heating station temperature" выбирается опция, соответствующую используемой конфигурации системы Firefly.

Если доступна опция "PREHT", то в разделе «Температура установки предварительного нагрева» "Pre-Heating Station Temperature" необходимо указать режим поступления температуры от внешнего устройства из «Автоматическое получение» "Automatic Acquisition", «Установить значение» "Set Value" и «Деактивировать» "Disable"[1]. Далее необходимо нажать клавишу «Создать программу» "Create Program" чтобы вызвать фазу исполнения создания программы пайки, которая заканчивается появлением отчета, содержащего информацию о выполненном действии. Клавиша «Следующая» "Next" позволяет перейти к следующей странице, где производится импорт CAD/CAM данных платы для пайки. Следующим этапом является открытие графического диалогового окна для выбора паяемого соединения. Каждому выбранному соединению можно привязать компонент, условие отказа от пайки, отмену, выбор других атрибутов (сторону пайки, форму и т.д.). Перемещая встроенную камеру системы оператор производит проверку и отладку настройки платы (координаты паяльных точек пайки, определение эталонных точек фиксации или изменение трансляционных точек, добавление новых пинов и т.д.). Для выполнения программы пайки используется страница "RUN", где происходит запуск пайки по заданной программе.

Программное обеспечение VIVA позволяет хранить температурный профиль для каждого паяного соединения, зафиксированные температуры и видеозаписи пайки. Таким образом, можно проверять производимые образцы и возможные ошибки в любое время и вносить соответствующие изменения[2].

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Руководство пользователя по эксплуатации «Система селективной пайки Firefly», 2016. С. 57.

2. Ефанов В. Лазерная технология селективной пайки. Достоинства и недостатки. [Электронный ресурс] / Мальцев П. // Печатный монтаж. 2010. – №2. – С.38-40. — Режим доступа: <http://circuitry.ru/journal/article/2190>.

УДК 006.015.7

К.А. ДАВЫДКИНРязанский государственный радиотехнический университет
имени В.Ф. Уткина**МОДЕЛЬ НАГРУЗКА-ПРОЧНОСТЬ ПРИ УСКОРЕННЫХ
ИСПЫТАНИЯХ НА НАДЕЖНОСТЬ**

В статье рассматривается модель нагрузка-прочность при испытаниях на надежность ускоренным методом.

Введение и постановка задачи. Современные изделия играют очень важную роль для решения сложных и важных задач, в связи с этим встает вопрос о более высоких требованиях к их надежности. Испытания на надежность проводятся на различных стадиях жизненного цикла изделий с целью установления их соответствия требованиям нормативно технической документации. Чтобы провести такие испытания потребуются большие затраты времени и средств.

Чтобы исключить вышеперечисленные проблемы существуют так называемые ускоренные испытания на надежность. Достигаются они путем увеличения воздействий испытательных факторов таких, как повышение температуры, токовых ударов включения или выключения питания. Перенос результатов ускоренных испытаний на нормальные условия эксплуатации требует соответствующей ускоряющей модели. Целью данной работы является применение модели нагрузка-несущая способность в качестве известной теории надежности [1, 2].

1. Расчетная модель. Если учитывать, что распределение прочности можно описать при помощи таких характеристик, как плотность вероятности $f_p(x)$, математическим ожиданием m_p и средним квадратичным отклонением σ_p , а распределение нагрузки – аналогичной функцией $f_\psi(x)$ с математическим ожиданием m_ψ и средним квадратичным отклонением σ_ψ , то вероятность безотказной работы (ВБР) узла определяется зависимостью [1]:

$$P = \Phi^* \left(\frac{m_p - m_\psi}{\sqrt{\sigma_p^2 + \sigma_\psi^2}} \right), \quad (1)$$

где Φ^* - нормальная функция распределения или интеграл вероятностей, для которого составлены таблицы [1, 3].

$$\Phi^*(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z e^{-\frac{z^2}{2}} dz \quad (2)$$

Здесь $z = (x-m) / \sigma$.

В практических расчетах надежности по модели «нагрузка-несущая способность» квантиль распределения $z = U_p$ можно выразить при помощи показателя коэффициента запаса прочности $n = m_p / m_\psi$. Тогда выражение (1) можно записать в виде

$$P = \Phi^* \left(\frac{n-1}{\sqrt{n^2 v_\rho^2 + v_\psi^2}} \right), \quad (3)$$

где v_ρ, v_ψ – коэффициенты вариации прочности и действующей нагрузки соответственно; $v = \sigma/m$.

Для утяжеленного режима испытаний [2] математическое ожидание прочности обозначим, как m'_ρ , несущую способность обозначим соответственно, как m'_ψ , а средние квадратичные отклонения прочности и действующей нагрузки соответственно σ'_ρ и σ'_ψ ; с учетом данных обозначений зависимость ВБР (1) будет выглядеть следующим образом:

$$P = \Phi^* \left(\frac{m'_\rho - m'_\psi}{\sqrt{(\sigma'_\rho)^2 + (\sigma'_\psi)^2}} \right), \quad (4)$$

в этом случае запас прочности можно выразить отношением

$$n' = \frac{m'_\rho}{m'_\psi}. \quad (5)$$

Степень утяжеления режима испытаний охарактеризуем коэффициентом

$$\eta = \frac{n}{n'} = \frac{m_\rho \cdot m'_\psi}{m_\psi \cdot m'_\rho}. \quad (6)$$

Если учитывать, что в процессе испытаний увеличивают только нагрузку, а прочность объекта испытаний не меняется, то коэффициент форсирования режима определяется как

$$\eta = \frac{n}{n'} = \frac{m'_\psi}{m_\psi}. \quad (7)$$

С учетом формулы (6) выражение вероятности безотказной работы получит следующий вид:

$$P = \Phi^* \left(\frac{n - \eta}{\sqrt{(n \nu'_\rho)^2 + (\eta \nu'_\psi)^2}} \right) \quad (8)$$

Из сопоставления выражений ВБР (8) и (3) видно, что при $\eta > 1$ (для случая неизменной прочности это получается при $m'_\psi > m_\psi$) значение квантиля распределения и сама функция Φ^* уменьшаются. Таким образом, можно для заданного коэффициента форсирования режима подчитать соответствующее значение ВБР, которое окажется меньше значения ВБР при эксплуатационной нагрузке. На основе данного вывода можно строить планы контрольных испытаний изделий с сокращением объема выборки или с уменьшением времени проведения. Рассмотрим подробнее.

2. Сокращение объема выборки. Известно, что в контрольных испытаниях на надежность устанавливают соответствие (или несоответствие) выпускаемой партии изделий объемом N заданному уровню надежности по результатам испытаний некоторой выборки [4] объемом n . При этом чем выше надежность, тем больший объем выборки требуется для подтверждения этого уровня надежности. Так, согласно формуле, связывающей объем выборки n , риск изготовителя α и приемлемо высокое значение вероятности безотказной работы P_1 для случая бездефектных испытаний (приемочное число равно нулю)

$$P_1 = \sqrt[n]{1 - \alpha} \quad (9)$$

меньшему значению n соответствует и меньшая вероятность P_1 .

При $\alpha = 0,1$ это соответствие значений P_1 и n получается следующим (табл. 1):

Таблица 1. Соответствие объема выборки подтверждаемой ВБР

n	3	5	10	20
P_1	0.46	0.63	0.79	0.89

Формула (9) выводится из более общей формулы биномиального распределения

$$C_n^0 \cdot (1 - P_1)^0 \cdot P_1^n = 1 - \alpha; \quad (10)$$

с учетом того, что число сочетаний из n по 0 равно 1.

Согласно указанной закономерности искусственное снижение ВБР изделия по сравнению с эксплуатационными значениями за счет повышения прикладываемой нагрузки позволяет уменьшить объем выборки при контрольных испытаниях на надежность. Численные значения, необходимые для составления плана испытаний, устанавливаются при этом формулами (5) – (10).

3. Экстраполяция результатов утяжеленных испытаний объекта с понижающей прочностью на отдаленное время эксплуатации. Метод, рассмотренный в п. 2 для сокращения объема выборки при испытаниях разрешается применять в случаях, когда прочностные характеристики изделий и прикладываемые нагрузки можно считать не изменяющимися во времени. Однако стоит учитывать, что в реальности данные величины будут зависеть от времени, и этот фактор чаще всего приходится учитывать.

Если задаваемую в процессе испытаний нагрузку можно контролировать, то прочность во времени проконтролировать невозможно. Обычно имеет место снижение прочности изделия ввиду его старения. Если учитывать, что с течением времени среднее значение прочности постепенно уменьшается, то для получения одинаковых квантилей работы объекта в настоящее время и в некоторое отдаленное время, достаточно увеличить нагрузку, но при условии сохранения всех средних квадратичных отклонений:

$$m_{\rho 0} - m_{\psi 0} = m_{\rho 0} - \Delta(t) - m_{\psi \text{раб}}, \quad (11)$$

где $m_{\psi 0}$ - нагрузка при испытаниях; $m_{\psi \text{раб}}$ - нагрузка при работе в отдаленный момент времени t .

Если модель деградации прочности известна функцией $\Delta(t)$, то из формулы (11) получается:

$$m_{\psi 0} = m_{\psi \text{раб}} + \Delta(t) \quad (12)$$

Приложение к объекту при испытаниях нагрузки с повышенным средним значением, подсчитанным по формуле (12), дает ту же вероятность отказа, что будет при нормальной нагрузке и уменьшенном значении прочности в отдаленный момент t . Если рассматривать изменение прочности как случайный процесс, то при известных характеристиках этого процесса можно на основе приведенных в п.1 формул строить модель ускорения испытаний.

Пересчет значения времени безотказной работы испытываемого изделия при повышенной нагрузке будет возможен при условии, что основные характеристики случайного процесса прочности ρ (временные зависимости - математического ожидания и дисперсии) будут известны.

Естественно принять как условие переноса результатов испытаний в утяжеленном режиме на нормальные эксплуатационные нагрузки равенство вероятностей безотказной работы в том и другом режимах. Время t , на которое экстраполируются результаты испытаний, отсчитывается от момента приложения испытательной утяжеленной нагрузки.

Если приравнять квантили функций (1) и (4), то сможем найти величины испытываемой нагрузки от времени эксплуатации. При этом в этих формулах математическое ожидание прочности во время утяжеленного испытаний и при эксплуатации связаны соотношением

$$m_{\rho} = m'_{\rho} - \Delta(t) \quad (13)$$

Также должны быть известны входящие в знаменатели выражений приравниваемых квантилей средние квадратичные отклонения прочности и нагрузки в том и другом режимах. Если учесть, что нагрузка при испытаниях может быть задана достаточно точно, чтобы пренебречь разбросом её значений ($\sigma_{\psi} \approx \sigma'_{\psi} \approx 0$), то получаем уравнение

$$\frac{m_{\rho} - m_{\psi}}{\sigma_{\rho}} = \frac{m'_{\rho} - m'_{\psi}}{\sigma'_{\rho}}; \quad (14)$$

отсюда, с учетом (11) при известной функции $\Delta(t)$ получается зависимость среднего значения m'_{ψ} утяжеляющей нагрузки от времени эксплуатации t , для которого ВБР $P(t)$ изделия можно считать равной ВБР $P(t=0)$, найденной в режиме утяжеленных испытаний.

В простом случае линейной модели деградации прочности $m_{\rho} = m'_{\rho} - kt$ и равенства средних квадратичных отклонений $\sigma'_{\rho} = \sigma_{\rho}$; $\sigma'_{\psi} = \sigma_{\psi}$ связь математических ожиданий испытательной и эксплуатационной нагрузок от времени получается следующей $m'_{\psi} = m_{\psi} + kt$.

Заключение и выводы. Таким образом, можно сделать вывод, что применение известной в теории надежности модели нагрузка-прочность позволит рассчитать нагрузку, задаваемую при ускоренных испытаниях, а также сократить объем испытаний.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Шишмарев В.Ю. Надежность технических систем : учебник для студ. высш. учеб. заведений / В.Ю. Шишмарев. — М. : Издательский центр «Академия», 2010. — 304 с.
2. Взаимозаменяемость и надежность/ (В.П. Федоров) // Рязань: Рязанский государственный радиотехнический университет им. В.Ф. Уткина, 2019. — 99с.
3. Чеканов А.М. Расчеты и обеспечение надежности электронной аппаратуры / А.М. Чеканов. — М. : издательство Кнорус, 2016— 438 с.
4. ГОСТ Р 27.403-2009 Надежность в технике (ССНТ). Планы испытаний для контроля вероятности безотказной работы.

УДК 62-712.2

А.Д. ДАНИЛОВ, В.В. СУСКИН

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

СТРУКТУРА И ПРИНЦИП РАБОТЫ СИСТЕМЫ ОХЛАЖДЕНИЯ НА ОСНОВЕ ТЕРМОЭЛЕКТРИЧЕСКИХ МОДУЛЕЙ

Приводится описание структуры и принципа работы термоэлектрической системы охлаждения, а также анализ существующих аналогов.

Термоэлектричество.

Под термином «термоэлектричество» подразумевается совокупность термических и электрических явлений, разница температур в которых создает электрический ток, или же электрических ток осуществляет возникновение разницы температур. [1]

Использование термоэлектрических устройств в системах охлаждения в последнее время набирает все большую популярность, область применения их с каждым годом становится всё шире. С ростом коэффициента полезного действия (КПД) термомодулей увеличивается холодопроизводительность систем, основанных на данной технологии. Помимо портативных холодильных камер, стационарных кулеров и охладителей электрических узлов, термоэлектрические системы применяются также для термостабилизации работы крупногабаритных блоков.

Термоэлектрические системы охлаждения, в основе работы которых заключается применение эффекта, открытого Жаном Шарль Пельтье в 1834 году, могут уменьшить выбросы вредных веществ в атмосферу, разрушающих озоновый слой Земли. Такими веществами являются гидрохлорофтороуглероды и хлорофтороуглероды – и посей день представляющее опасность, поскольку были долгое время основными компонентами в области технологий охлаждения.

Интерес в исследовании данной области обусловлен во многом благодаря разворачивающейся политике мировых стран в сторону «зеленой» энергетики, то есть в направлении развития альтернативных источников энергии, где для термоэлектричества отводится важная роль. Во многих странах принимаются законопроекты, регулирующие применение вышеуказанных химических соединений для охлаждения, и использования их на производстве в целом. На данный момент на

законодательном уровне контролируются объёмы этих химических веществ, постепенно исключается возможность их производства: с 2021 года в развитых странах, а с 2031 в развивающихся странах.

Данные меры и обеспокоенность состоянием окружающей среды, экологическая ситуация в мире стимулируют рост развития технологий в области термоэлектричества. Совершенствование технологий «зеленого» сектора приведет к появлению эффективных термоэлектрических охлаждающих элементов. Такие элементы помогут сократить выбросы в атмосферу вредных химических веществ и уменьшить звуковое загрязнение окружающего пространства (поскольку они являются твердотельными элементами и не нуждаются в шумных компрессорах).

Несмотря на то, что технология термоэлектричества известна уже около трех веков, ощутимый рост в развитии и интерес в государственных масштабах она приобрела относительно недавно, поэтому парокомпрессионные охладители все еще более эффективны, чем охладители Пельтье, но они очень громоздкие, поэтому занимают слишком много полезной площади, ко всему прочему сложны в обслуживании и имеют низкую ремонтопригодность.

Устройство охлаждения.

При рассмотрении систем охлаждения, нацеленных на вентилирование воздушным потоком (с более низкой температурой, чем температура окружающей среды) отдельных блоков, стоек или целых комплексов, следует отметить, что в настоящий момент наиболее широко применяются системы, в основе которых лежат методы компрессии и конденсации хладагента (в основном - фреона). Устройства данного типа имеют структуру из двух основных блоков (контуров): наружного блока и внутреннего. Принцип работы сплит-систем заключается в охлаждении и нагреве хладагента, циркулирующего в замкнутой системе двух контуров.

Как уже указывалось ранее, данные системы крупногабаритные, работа компрессора сопровождается шумом, а хладагент при его попадании в атмосферу, помимо разрушения защитного слоя атмосферы, также приводит к усилению отрицательного воздействия парникового эффекта, от губительных последствий которого уже сейчас все мировое сообщество интенсивно пытается найти решение.

Из всего вышеуказанного следует выделить две основные идеи, на которых основывается описываемая в данной статье разработка:

1. Получение минимальных габаритных размеров устройства охлаждения, достижение его высокой надежности и ремонтопригодности (за счет твердотельной технологии),

мобильности, а также сокращение создаваемого шума при эксплуатации;

2. Разработка устройства, обеспечивающего отсутствие негативного воздействия на окружающую среду.

Ранее были изучены основные преграды в развитии термомодулей и увеличении их КПД, в основе которых – низкое значение показателя термоэлектрической добротности ZT и современные, инновационные предложения по осуществлению контакта полупроводников внутри модуля. Но по последним данным и все большему числу публикуемых научных работ посвящённых данной тематике, наблюдаются значительные успехи в решении данных задач: открытие новых полупроводниковых материалов, а в качестве связующего элемента применение жидких соединителей взамен жестких пластин. Всё это дает возможность в реализации высокоэффективных, высокотехнологичных и экологически чистых устройств охлаждения. [2]

В данной статье рассматривается устройство, которое обеспечивает подачу воздушного потока в блок путем автоматического управления и поддержания температуры воздушного потока релейного действия и с обратной связью. Управляющие воздействия, в данном случае, дискретного типа. Данная система будет построена на основе термоэлектрических модулей, производимых компанией «Криотерм». По сравнению с классическим, на данный момент, применением модулей на их основе будет спроектирована система охлаждения с криогенной мощностью не менее 1 кВт. В устройстве обдува будет предусмотрена световая индикация достижения заданной температуры охлаждения.

Рассматриваемая система осуществляет подачу охлажденного воздуха в объекты контроля (ОК), а также позволяет поддерживать заданную температуру на протяжении всего времени проверки ОК. Устройство имеет следующие составные части:

1. Основной блок системы охлаждения – охладительный канал, в котором располагаются термоэлектрические модули (ТЭМ). Для отвода тепла с горячей стороны модулей устанавливаются радиаторы из материала с высокой теплопроводностью. В качестве теплообменника между воздушным потоком и холодной стороной ТЭМ также используется радиатор. Для регистрации температуры воздуха на выходе устройства, в конце канала устанавливается ячейка с чувствительным элементом, в качестве которого выступает терморезистор.

2. Узел управления – часть системы, которая отвечает за регулировку температуры на выходе устройства. Данный узел регистрирует показатели с чувствительного термоэлемента, и в зависимости от этих значений осуществляет коммутацию термомодулей, подключая или отключая их.

3. Блок питания системы – питающее устройство, которое обеспечивает работу всей системы в целом, генерирует электрическую энергию для функционирования элементов Пельтье и нагнетателя.

4. Вентилятор - нагнетатель воздушного потока в канал охлаждения. Он обеспечивает на выходе устройства требуемые значения расхода и давления воздуха, а также необходим для преодоления гидравлического сопротивления на входе в ОК.

Теперь следует привести краткое описание принципа работы всей системы. Из внешней среды (комнаты или кузова-контейнера) вентилятор всасывает воздушный поток и нагнетает его через распределительный растроб во входное отверстие канала охлаждения. Датчик температуры регистрирует параметры воздушного потока на выходе устройства и, в зависимости от его показаний, блок управления осуществляет подключение ТЭМ к системе питания. До того момента, когда температура достигнет установленного значения, будет гореть индикатор «Т не в норме», по достижению - «Т в норме». В зависимости от температуры воздуха окружающей среды будет требоваться разное значение холодопроизводительности устройства. За количество подключённых термоэлементов будет отвечать узел управления. Принцип охлаждения воздушного потока в канале устройства основан на свойстве элементов Пельтье выступать в качестве теплового насоса. [3] Со стороны канала охлаждения располагаются радиаторы, связанные с холодной стороной ТЭМ. Эти радиаторы поглощают тепловую энергию воздушного потока и передают ее на модули, которые перекачивают энергию с холодной пластины на горячую, откуда тепло рассеивается благодаря внешним радиаторам.

Главной задачей при проектировании описываемых устройств будет являться наиболее полный теплофизический расчет и анализ системы, которые должны отражать реальную картину следующих параметров: необходимую холодопроизводительность, тип и количество термоэлектрических модулей, необходимость дополнительных средств отвода тепла с горячей стороны ТЭМ-ов для поддержания максимально возможной разницы температур между сторонами термомодулей, что обеспечит наивысшую эффективность их работы, достижение максимально возможного равномерного

контакта по всей площади соприкосновения поверхностей радиаторов и ТЭМ-ов для наилучшего теплоотвода. Также важным моментом будет являться решение о выборе конструкционных материалов, которые должны обеспечить высокую степень термо- и электроизоляционных свойств, а также жесткость и прочность конструкции.

Более подробно данная конструкция будет описана в дальнейших работах на эту тему.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Данилов А.Д., Сускин В.В. Термоэлектрические устройства охлаждения / Информационные технологии, межвузовский сборник научных трудов. – Рязань: ИП Коняхин А.В. (Book Jet), 2020 – с. 38 - 42.

2. G.D. Mahan. Good Thermoelectrics// Solid State Physics. 1998. Vol. 51, pp. 81–157.

3. Улитенко А.И., Гуров В.С., Пушкин В.А. Принципы построения индивидуальных систем охлаждения электронных приборов и устройств. – М.: Горячая линия–Телеком, 2012. – 286 с.: ил.

УДК 004.75

И.В. ДУРАЕВРязанский государственный радиотехнический университет
имени В. Ф. Уткина**АНАЛИЗ СОВРЕМЕННЫХ ПРИЕМОПЕРЕДАЮЩИХ
ИНТЕРФЕЙСОВ***Рассматриваются различные способы коммутации с внешними устройствами посредством аппаратных интерфейсов.*

Аппаратным интерфейсом принято называть набор правил унифицированных взаимоотношений между отдельными радиоэлектронными устройствами, а также ряд аппаратных, схмотехнических и конструктивных решений, позволяющих реализовать эту систему. Обычно в коммуникационной среде вычислительных устройств используется широкий спектр аппаратных и программных интерфейсов различных типов и рангов. Связь между продуктами осуществляется посредством интерфейсных линий с использованием сигналов. В рамках данной статьи мы разберем основные типы интерфейсов, используемых в современных радиоэлектронных средствах.

CAN интерфейс.

Последовательный интерфейс CAN (Controller Area Network) - это интерфейс связи, который может эффективно поддерживать распределенное автоматическое управление в реальном времени в среде с высокой помехоустойчивостью. Протокол связи определен в спецификации CAN 2.0B. Протокол CAN активно используется в различных сферах, начиная от высокоскоростных сетей связи и заканчивая электропроводкой в транспортных средствах. Достаточно высокая скорость передачи данных (до 1 Мбит / с), приемлемая помехоустойчивость протокола, защита от сбоя схем и структурных узлов - делают CAN-шину наиболее рационально подходящей для приложений промышленного управления, таких как Device Net.

Обычно стандартный узел CAN обычно состоит из микроконтроллера, контроллера CAN и приемопередатчика. В последнее время появилась тенденция использовать микроконтроллеры со встроенным CAN-контроллером для упрощения схемы, однако есть автономный миниатюрный CAN-контроллер с интерфейсом SPI (например, MCP2510). Далее трансивер подключается к витой паре, на концах которой установлены

согласующие резисторы (терминатор) сопротивлением 120 Ом. В протоколе CAN при передаче данных принимающие узлы не адресуются, но указывается идентификатор передатчика. Идентификатор указывает содержание сообщения (например, в случае автомобиля - обороты, температура двигателя и т. Д.) И степень приоритета сообщения. Более высокий приоритет отдается идентификатору с меньшим двоичным значением.

Преимущества:

- умение работать в режиме реального времени;
- высокая устойчивость к помехам;
- надежный контроль ошибок передачи и приема.

Недостатки:

- максимальная длина сети обратно пропорциональна скорости передачи;
- большой размер служебных данных в пакете[1].

Интерфейс Ethernet.

Технология Ethernet позволяет использовать различные среды передачи, для каждой из которых существует стандартное имя, например XBaseY, где X - скорость передачи, Мбит / с (10, 100, 1000 ...); База - ключевое слово (обозначает модулированную передачу); Y обозначает среду передачи и дальность связи. Во всех современных версиях Ethernet используется витая пара или оптоволоконная или звездообразная топология. Центральным устройством звезды может быть повторитель (он же концентратор) или коммутатор.

Для оптических подключений используются различные разъемы: ST, SC, MT-RJ и другие. Обратите внимание, что каждый сегмент подключения должен быть завершен терминаторами 50 Ом и заземлен в одной точке. «Заземление цепи» устройства не гальванически связано с экраном коаксиального разъема, поэтому избегайте случайного прикосновения к разъемам BNC металлическими частями, подключенными к шасси. Сети на коаксиальных соединениях требуют правильного заземления, нарушение правил чревато выгоранием переходников. Порты на концентраторах 10BaseT, 100BaseTX и 100BaseT4 являются портами MDIX. Для подключения конечных узлов к портам активного оборудования используется «прямой» кабель для прямого подключения адаптеров или «перекрестный» кабель для подключения двух устройств связи.

Преимущества:

- простота установки;
- высокая пропускная способность и скорость передачи данных;

Недостатки:

- снижение реальной скорости передачи данных в сильно загруженной сети до ее полной остановки из-за конфликтов в среде передачи данных;

- сложность устранения неисправностей[2].

Интерфейс USB.

Обычная архитектура шины USB предполагает подключение одного или нескольких USB-устройств к компьютеру. Компьютер в этой конфигурации является основным управляющим устройством и называется хостом. Устройства подключаются к хосту с помощью кабелей. В компьютере есть встроенный концентратор, называемый корневым концентратором. Информационные сигналы и напряжение питания передаются по четырехжильному кабелю. В шине используется дифференциальный метод передачи сигналов D + и D- по двум проводам. Сигналы синхронизации и данные кодируются с использованием метода NRZI. В этом кодировании логическая «1» представлена постоянным уровнем на протяжении битового интервала, а логический «0» - это изменение уровня на противоположный в течение битового интервала. Дифференциальный "0" передается путем подтягивания линии D + к напряжению более 2,8 В.

Преимущества:

- распространенность устройств;
- высокая пропускная способность;

Недостатки:

- интерфейс не предназначен для создания локальных сетей;
- ток питания ограничен величиной порядка 0,5 А[3].

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Интерфейсы периферийных устройств : учеб. пособие для вузов / В. А. Павлов. — Саратов, 2010. — 374 с.:

2. Ключев А.О., Ковязина Д.Р., Петров Е.В., Платунов А.Е. Интерфейсы периферийных устройств. – СПб.: СПбГУ ИТМО, 2010. – 290 с

3. Интерфейсы USB. Практика использования и программирования / Агуров П. В. – СПб.: БХВ-Петербург, 2004. – 576 с.

УДК 062.69

И.В. ДУРАЕВРязанский государственный радиотехнический университет
имени В. Ф. Уткина**ИССЛЕДОВАНИЕ ВОЗМОЖНОСТЕЙ ПРИМЕНЕНИЯ
ПОДОГРЕВА РЭА***Рассматриваются различные способы подогрева внутрикорпусного пространства радиоэлектронной аппаратуры в условиях низких температур.*

Главным способом поддержания стабильной работы устройства в условиях низких температур является устройство системы подогрева. Подразумевается, что посредством температурного датчика будет отслеживаться температурный режим с подачей соответствующих данных на микроконтроллер. В случае снижения температуры ниже установленной отметки, микроконтроллер подает управляющее напряжение на нагревательный элемент, который должен вернуть температурный режим в допустимый предел работы устройства. Рассмотрим основные варианты нагревательных элементов.

Гибкая нагревательная пластина.

Греющая пластина представляет собой гибкий и эластичный мат из литого теплопроводящего изолятора, в котором запаяны провода. Благодаря гибкости нагревающие пластины могут фиксироваться на поверхностях произвольных геометрических форм и размеров. Главное условие – отсутствие ребер жесткости и выступов на гладкой поверхности. Изделия поставляются с надежно клеящейся основой. Гибкие нагревающие силиконовые пластины применяют для обогрева ходовой части транспорта, аэрокосмического оборудования и в медицине.

Нагреватель имеет трехслойную структуру. Верхний слой и нижний – силикон, армированный стекловолокном. Это позволяет защитить нити от переломов при монтаже нагревателя на изогнутые поверхности. Средний слой – нити из сплава вольфрама и никрома. Нагреватель имеет высокую теплоотдачу за счет быстрого нагрева металлических нитей и тонкого слоя силикона. В состав также входят соединительные провода и термодатчик. Основным критерием его надежности и долговечности считается однородность силиконового материала, то есть отсутствие пор и пустот в его структуре. Это позволяет избежать локального перегрева внутри самого нагревателя. Нагреватель может крепиться на высокотемпературный монтажный

клеевой слой на одной из сторон нагревателя с предельной температурой 150° С; при отсутствии монтажного элемента – на силиконовый высокотемпературный герметик; на текстильную застежку; через отверстия с металлическим укреплением; на алюминиевую монтажную панель.

Основные преимущества использования:

1. Прямой монтаж с внутренней стороны на монтажную панель за счет наличия высокотемпературного слоя;

2. Защита от пробоя на корпус, благодаря размещению нагревателя среди слоев силикона;

3. Любое напряжение, возможность изготовления нагревателей любой мощности исходя из технического задания [1].

Тонкопленочные резисторы с напылением.

Это новшество относится к области микроэлектроники. Тонкопленочный резистор содержит в себе резистивный элемент прямоугольной формы и два электрода из многослойной проводящей структуры, имеющей гребенчатую форму и некоторое количество резистивных элементов, которые расположены параллельно ширине электродов, и $n-1$ окон прямоугольной формы, свободных от пленочных элементов и которые расположены при окончании резистивных элементов под углом 90 градусов к ним, а коэффициент формы каждого элемента варьируется в диапазоне от 0,7 до 0,13, при этом совокупная длина электродов должна быть не менее длины резистивных элементов. Принцип изготовления тонкопленочного резистора заключается в напылении на керамическую подложку резистивного слоя и многослойной проводящей структуры, и проведения двойной фотолитографии, при этом после первой фотолитографии травлением многослойной проводящей структуры формируются встречно-гребенчатые электроды, второй фотолитографии формируются резистивные элементы, при второй фотолитографии фоторезист наносится на весь резистор за исключением окон, ширина окон равна минимальной ширине зазора между электродами, который достигим технологически, после чего в окнах производят травление резистивного слоя. Техническим результатом является расширение диапазона сопротивлений ТПР, изготавливаемых на одной плате [2].

Применение тепловой мощности транзистора.

Другим способом организации нагрева является применение тепловой энергии транзистора с включением по схеме генератора стабильного тока (рисунок 1).

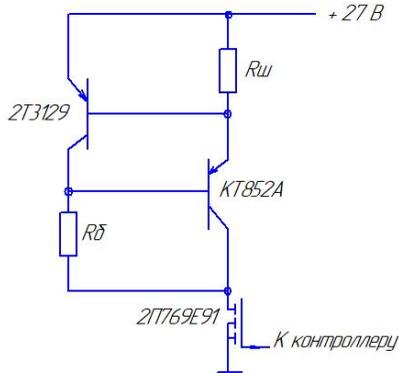


Рисунок 1 – Включение транзистора по схеме генератора стабильного тока

Особенность заключается в том, что выходной ток не зависит от напряжения питания. В этой схеме напряжение $U_{6Э}$ транзистора VT_1 падая на резисторе $R_{ш}$, определяет выходной ток независимо от напряжения $U_{КК}$

$$I_K = U_{6Э} / R_{ш} \quad (1.1).$$

С помощью резистора $R_б$ устанавливается смещение транзистора VT_2 и потенциал коллектора VT_1 , причем этот потенциал меньше, чем напряжение $U_{КК}$, на удвоенную величину падения напряжения на переходе; тем самым уменьшается влияние эффекта Эрли. Постоянное напряжение с микроконтроллера управляет нагревом [3].

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Кееново – гибкие греющие силиконовые пластины [Электронный ресурс] – Режим доступа: <http://keenovo.ru>.
2. Исследование влияния электрических, технологических и эксплуатационных характеристик на конструктивные параметры тонкопленочных резисторов: метод. указания/сост. А.И. Меркулов. – Самара: Изд-во Самар. гос. аэрокосм. ун-та, 2010. – 28 с.
3. Хоровиц П. Хилл У. Искусство схемотехники: Пер. с английского Изд. 2-е – М.: Издательство БИНОМ. – 2014. – 704 с., ил. УДК 004.9

УДК 621.374.3

А.В. ЖИЛЬЦОВРязанский государственный радиотехнический университет
имени В.Ф. Уткина**ПЕРЕКЛЮЧАТЕЛЬ СОСТОЯНИЯ НА ОСНОВЕ МОДЕЛИ Т-ТРИГГЕРА НА ПЛИС**

Рассматривается краткое описание ПЛИС, приводятся различия с микроконтроллерами. Также рассматривается простейший переключатель состояния на основе модели Т-триггера и то, как можно запрограммировать ПЛИС на основе данной модели.

Что такое ПЛИС и с чем связана цифровая электроника? На этот вопрос мы постараемся ответить в данной статье, а также рассмотрим модель Т-триггера, который является по сути переключателем состояния и запрограммируем ПЛИС на его работу, используя язык описания аппаратуры VHDL.

Начнём с того, что цифровая электроника ассоциируется прежде всего с логическими элементами И, ИЛИ, НЕ. Далее регистры сдвига, триггеры, дешифраторы, мультиплексоры и т.д. Однако, по мере роста сложности электронных устройств и устройств на основе вышеупомянутых компонентов, создание приборов на базе интегральных микросхем (ИМС) стало затруднительным, так как заказные ИМС нужной топологии и схемотехники оправдывали себя только при серийном производстве устройства, а в случае единичного использования это было довольно дорого. Выходом из сложившейся ситуации стала разработка программируемых интегральных логических схем (сокращенно ПЛИС) [1].

При упоминании слов «программируемый» у большинства людей или новичков в области программирования микросхем возникает ассоциация с микроконтроллерами. Отчасти это правда, они тоже программируются, но всё же ПЛИС это совсем другое устройство.

Для разработчиков в микроконтроллерах возможен фиксированный набор решений и средств, присущий конкретному чипу; нет возможности каким-либо образом отклоняться от архитектуры. Вам доступен набор команд, с помощью которых вы выполняете операции по взаимодействию с окружающей средой, считывая данные с помощью цифровых и аналоговых входов, и

отправляя сигналы другим исполнительным устройствам через выходные порты микросхемы.

Вы также можете выполнять вычисления, такие как, сохранение информации в регистрах или ПЗУ, а также производить различные операции с данными, записанными в память микроконтроллера. Собственно это и есть назначение и особенности работы с микроконтроллерами.

Основное различие между программируемыми логическими интегральными схемами (ПЛИС) заключается в том, что, запрограммировав устройство, вы сами создаете свою архитектуру из базовых логических элементов. Благодаря этому вы получаете высокую скорость работы и гибкость микросхемы. Это даёт возможность без изменения микросхемы делать множество различных проектов [1].

В целом внутреннюю структуру ПЛИС можно разделить на три основные группы: массив логических элементов (макроячейки, логические блоки); блоки ввода-вывода (IO); линии связи между ними, которое управляют этим соединениями.

Следует отметить, что это данная структура представлена в обобщенном виде. Программируя микросхему, вы соединяете элементы так же, как если бы вы собирали устройство из отдельных элементов и соединяли их входы и выходы проводниками.

Теперь разберёмся что такое триггер. Триггер это устройство, имеющее два устойчивых состояния, одно из которых соответствует двоичной единице, а второе двоичному нулю. В общем случае основная функция триггеров – запоминание двоичной информации. Использование триггеров позволяет реализовывать устройства оперативной памяти, т.е. памяти, данные в которой хранятся только в процессе работы устройства, его вычислений. Следует отметить, что это не единственная их отрасль применения. Триггеры также широко используются для создания цифровых устройств с памятью, например таких как счётчики, последовательные порты или цифровые линии задержки, применяемые в составе цифровых фильтров [2].

Рассматриваемый Т-триггер является одной из возможных разновидностей триггеров и является счётным триггером, также у него имеется один информационный вход и выход. После поступления на вход импульса, состояние триггера меняется на прямо противоположное. Счётным он является потому, что Т-триггер как бы считает количество импульсов, которые поступают на его вход. При поступлении на вход второго импульса Т-триггер опять сбрасывается в исходное состояние.

Рассмотрим двухступенчатый Т-триггер, который представлен на рисунке 1. Как видно из схемы, у данного триггера имеется один вход С, который является входом синхронизации, также имеется один выход Q. При поступлении на вход С двоичной единицы – значение на выходе Q инвертируется, при переходе значения сигнала на входе из двоичной единицы в ноль и при наличии на входе двоичного нуля – значение сигнала на выходе Q остаётся неизменным. Таким образом, данный триггер переключается по переднему фронту [3].

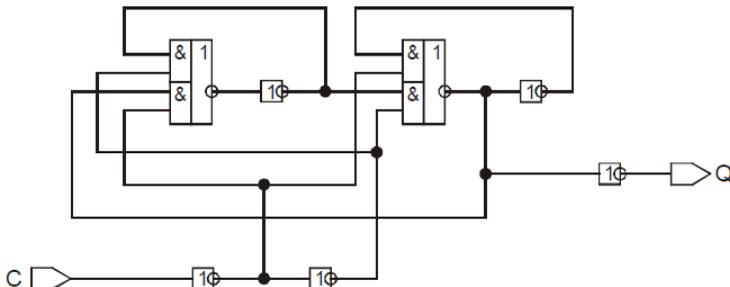


Рисунок 1 – Двухступенчатый Т-триггер

Про модулируем данную схему в программе ModelSim, которая является довольно мощной средой для симуляции и отладки программ языке VHDL [4]. Напишем для этого программу:

```
entity T_Func is
port (C : in BIT; Q : inout BIT);
end T_Func;
architecture functional of T_Func is
signal T:bit;
begin
p1 : process(C)
begin
T<= not Q; Q <= T;
end process p1;
end functional;
```

Напишем тестирующую программу и в последствие соединим вместе оба файла в один проект:

```
Entity test_T is
end test_T;
architecture beh of test_T is
component T_func
port (C : in BIT; Q : inout BIT);
end component;
signal C, Q : BIT;
begin
```

```

p2 : T_func port map (C, Q);
C <= '0',
'1' after 100 ns, '0' after 200 ns, '1' after 300 ns,
'0' after 400 ns, '1' after 500 ns, '0' after 600 ns,
'1' after 700 ns, '0' after 800 ns, '1' after 900 ns,
'0' after 1000 ns;
end beh;

```

Результат работы можно увидеть на диаграмме (рисунок 2):



Рисунок 2 – Временная работа диаграммы программы

Таким образом, как видно из временной диаграммы, триггер функционирует правильно. Теперь рассмотрим как полученные знания из работы в ModelSim можно применить на практике. Рассмотрим ПЛИС модели EP4CE6E22C8L (рисунок 3). ПЛИС программируется в программе Quartus II, которая представляет собой систему автоматизированного проектирования цифровых устройств на кристаллах ПЛИС, фирмы Altera также на языке VHDL [5]. Данная ПЛИС имеет 4 кнопки (PIN_89, PIN_90, PIN_91, PIN_88), светодиоды (PIN_141, PIN_1, PIN_2, PIN_3). Кнопки являются не фиксирующимися, и при нажатии не переключаются, т.е. меняют своё значение сигнала только в момент нажатия, а далее, после отпускания кнопки не сохраняют текущее значение. Опирируя этими данными, запрограммируем данную ПЛИС на работу Т-триггера.

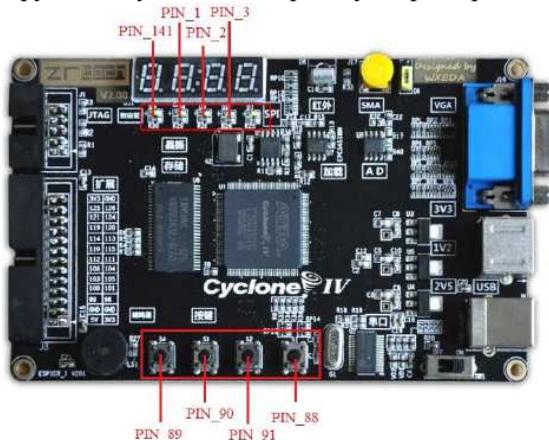


Рисунок 3 – ПЛИС

```
Программный код:
package tt is
  procedure trigger_proc(signal beforeQ: inout bit;
signal Q: out bit );
  end tt;
  package body tt is
    procedure trigger_proc(signal beforeQ: inout bit;
signal Q: out bit) is
      begin
        Q<= not beforeQ; Before Q<= not beforeQ;
      end procedure;
    end tt;
  library work; library ieee;
  use ieee.std_logic_1164.all;
  use work.tt.all;
  entity main is
    port (CLK: in bit; R: in bit; Q: out bit);
  end main;
  architecture arch of main is
    component Trigger
    port (CLK: in bit; R: in bit; Q: out bit);
  end component;
  begin
    x1: Trigger port map(CLK, R, Q);
  end arch;
  library work;
  use work.tt.all;
  entity Trigger is
    port (CLK: in bit; R: in bit; Q: out bit);
  end Trigger;
  architecture arch of Trigger is
    signal tempQ: bit := '1';
    begin process(R)
      begin
        if (R = '0') then trigger_proc(tempQ,Q);
        end if;
      end process;
    end arch;
```

В данной программе в качестве синхросигналов С выступает кнопка (PIN_90), а в качестве демонстрирующего работу данного Т-триггера – светодиод (PIN_3). Принцип работы программы заключается в том, что при нажатии кнопки светодиод сначала загорается, а затем, когда отпускается кнопка, текущее значение сигнала сохраняется в памяти ПЛИС и светодиод продолжает гореть, при повторном нажатии гаснет. Таким образом, функционирование

ПЛИС по данной программе отражает суть работы Т-триггера. При нажатии кнопки, т.е. при переходе из двоичного состояния ноль в состояние двоичной единицы, светодиод загорается, т.е. сначала на выходе был сигнал равный нулю (на светодиоде). Затем когда отпускается кнопка, т.е. при переходе в состояние из единицы в ноль светодиод продолжает светиться, т.е. сохраняет своё предыдущее значение. После повторного нажатия кнопки значение инвертируется и светодиод гаснет. Как видно из работы данного триггера, запрограммированного на ПЛИС, данная модель является по сути «включателем» или «выключателем», например светодиода, это явно отражено в анализе работы ПЛИС по данной программе.

Таким образом было проведено моделирование Т-триггера с помощью программы ModelSim, а затем в программе Quartus II. Основу работы Т-триггера можно использовать например для создания новогодней гирлянды, если вместо кнопки подать на вход синхроимпульсы CLK и изменив соответствующим образом программный код, получим то, что на выходе светодиода будет моргать; можно также для больше наглядности потом будет подключить все оставшиеся светодиоды, или при желании подключить светодиодную ленту через выходы микросхемы. Также данная модель Т-триггера на ПЛИС может иметь полезную функцию, например когда есть необходимость активировать какой-либо процесс на «постоянной основе», т.е. с фиксацией и сохранением текущего значения после активации процесса нажатием кнопки.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Сапрыкин А.Н. Конспект лекций по курсу «Информационные технологии проектирования ЭВС».
2. Перепелкин, Д.А. Схемотехника усилительных устройств: Учебное пособие для вузов / Д.А. Перепелкин. - М.: Горячая линия - Телеком, 2014. - 238 с.
3. Триггеры: Методические указания к лабораторной работе №7 / Рязань. гос. радиотехн. ун-тет; Сост.: Сапрыкин А.Н. Рязань, 2020. 12 с.
4. Поляков А. К. Языки VHDL и VERILOG в проектировании цифровой аппаратуры. – М.: СОЛОН-Пресс, 2003.- 320 с.: ил. – (Серия «Системы проектирования»).
5. Грушвицкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем на микросхемах программируемой логики. – СПб.: БХВ-Петербург, 2002.- 608 с.

УДК 004.021

О.А. ЖУКОВАРязанский государственный радиотехнический университет
имени В.Ф. Уткина**ПРИМЕНЕНИЕ ПРОГРАММНЫХ СРЕДСТВ, РЕАЛИЗУЮЩИХ
АЛГОРИТМ ЭКСПЕРТНОГО ОЦЕНИВАНИЯ НА ОСНОВЕ
МЕДИАНЫ КЕМЕНИ**

В статье рассматривается актуальность метода экспертного оценивания для получения качественной информации. Так же рассматривается целесообразность применения вычислительной техники для обработки и анализа экспертных оценок с помощью вычисления медианы Кемени.

Устойчивое существование и развитие любой организации главным образом зависит от того, как этой организацией управляют, как подготавливают и принимают решения, определяющие ее деятельность. Проблему принятия решения можно рассматривать как проблему выбора одной альтернативы из множества возможных. Обоснование такого выбора связано с получением и анализом необходимой информации. Некоторые данные, характеризующие управляемую систему и условия ее функционирования, могут быть получены в результате непосредственных измерений. Другие данные измерить невозможно либо потому, что не существует соответствующих измерительных приборов, либо потому, что представляющая интерес информация является субъективной. Для оценки показателей, значения которых не могут быть непосредственно измерены, привлекаются опытные специалисты — эксперты [1].

Основная задача экспертного оценивания — задача определения значений качественного признака у объектов. Если задано некоторое число однородных объектов, характеризующихся общим качественным признаком, то на основе сравнения между собой объектов по этому признаку эксперт может выбрать наиболее предпочтительный из них, либо упорядочить все объекты по степени проявления признака, либо приписать каждому объекту количественную оценку, выражающую интенсивность проявления рассматриваемого качественного признака у этого объекта.

Экспертные оценки в настоящее время являются наиболее распространенным способом получения и анализа качественной информации. Поэтому к ним привлечено внимание широкого круга специалистов.

Возможности использования информации, полученной от специалистов, значительно расширяются с использованием методов экспертного оценивания. При выборе перспективных решений, даже самые простые статистические методы в сочетании с информацией, полученной от экспертов, приводят к более успешным результатам, чем точные расчеты, ориентированные на средние показатели [4].

Для достижения высокого качества получаемой экспертной информации, необходимо выполнение следующих условий:

- наличие комиссии экспертов, состоящей из специалистов и профессионалов хорошо знакомых с объектом экспертизы;
- наличие специалистов имеющих опыт проведения экспертиз;
- получение точной информации от экспертов;
- корректная обработка экспертной информации.

Использование экспертной информации особенно плодотворно, если для ее сбора, обработки и анализа применяются специальные программные средства реализующие алгоритмы экспертного оценивания. Так как обработка результатов экспертизы вручную вызывает большие трудовые затраты, то ее целесообразно проводить с помощью вычислительной техники.

Применение вычислительной техники на первый план выдвигает проблему разработки программных средств, которые реализуют обработку результатов экспертного оценивания. При организации обработки результатов опроса следует тщательно проанализировать трудоемкости решения задач с учетом разработки математического обеспечения для ЭВМ.

Для разработки прикладной вычислительной экспертной системы, оптимальным по вычислительной сложности является метод медианы Кемени. Для медианы Кемени справедлив закон больших чисел, который показывает что она обладает устойчивостью по отношению к незначительному изменению состава экспертной комиссии, и при увеличении числа экспертов она приближается к некоторому пределу, который может рассматриваться как истинное мнение экспертов, от которого каждый из них несколько отклонялся по случайным причинам[3].

Вычисление медианы - это задача целочисленного программирования. Существует большое количество способов поиска медиан Кемени, но чаще всего используются алгоритмы, основанные на методе ветвей и границ. Однако, работы российского ученого Жихарева В. Н. [2] показали, что алгоритм полного перебора, считается самым точным для поиска всех медиан для заданного множества перестановок. На современных компьютерах, при

относительно небольшом количестве экспертов, данный алгоритм может быть достаточно эффективно применен для поиска медиан Кемени.

Изучив алгоритмы разработанные Жихаревым, и проанализировав свойства медианы Кемени, можно сделать следующие выводы:

1. Сравнительно небольшая средняя величина количества медиан, но больше единицы
2. Количество медиан, иногда, может сильно превзойти количество экспертных ответов.
3. В большинстве случаев множество медиан пересекается с множеством экспертов.
4. В некоторых случаях медианы могут быть далеко удалены от множества экспертов.
5. Целесообразно широко применять медиану Кемени для решения различных прикладных задач.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Полякова Н.Л. XX век в социологических теориях общества. — М.: Логос, 2004.
2. Жихарев В. Н., Орлов А. И. Законы больших чисел и состоятельность статистических оценок в пространствах произвольной природы // Статистические методы оценивания и проверки гипотез: межвуз. сб. науч. тр. – Пермь: Изд-во Пермского госуд. ун-та, 1998. – С. 65-84.
3. Жихарев В.Н. Медиана Кемени. [Электронный ресурс] URL: <http://www.bmstu.ru/ps/%7Eorlov/> (дата обращения 13.04.2021).
4. Глушков В.М. О прогнозировании на основе экспертных оценок // Науковедение. Прогнозирование. Информатика. — Киев, 1970.

УДК 004.65

И.С. ЗАХАРОВ

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

НАСТРАИВАЕМЫЙ БЕСПИЛОТНЫЙ НАЗЕМНЫЙ АВТОМОБИЛЬ С ДИСТАНЦИОННЫМ УПРАВЛЕНИЕМ

Рассматривается настраиваемый беспилотный наземный автомобиль с дистанционным управлением и его роль, а также система управления UGV и возможность модификации данной системы.

В сегодняшнюю научную эпоху технологический прогресс стремительно набирает обороты. Лидер в этом марафоне за действительно красивое будущее, основанное на технологиях, — это система на базе Linux. Его философия открытого исходного кода позволил миру обуздать и извлечь выгоду из величайших умов мира. Это также сделало Linux наиболее доступной продвинутой операционной системой.

Разрабатывая приложения на базе Linux, можно создавать суперумные устройства, такие как беспилотные наземные машины. С пониманием Linux как инструмента мы поймем разработку легко настраиваемый беспилотный наземный автомобиль с дистанционным управлением.

Спрос на беспилотные наземные транспортные средства (UGV) неуклонно растет, поскольку они спасают людей от выполнения опасных и утомительных работ. UGV также будут играть важную роль для следующих поколений, например в сухопутных войсках. Несмотря на значительный технологический прогресс, полностью автономные внедорожники далеки от реальности. В настоящее время большинство UGV имеют дистанционное управление, управляемое людьми. Одним из главных преимуществ является управление UGV с относительно безопасного расстояния, отправив управление и навигацию команды по беспроводной сети.

Требования к оборудованию:

1. x86 / i386;
2. Система на базе ARM / MIPS;
3. Любые базовые периферийные устройства;
4. Кабели для связи, которые можно использовать.

Требования к программному обеспечению:

1. Дистрибутив Linux.;

2. Обычный исходный код ядра 2.6.xx для Vanilla Linux;
3. uClibc - Маленькая стандартная библиотека C для разработки встраиваемого Linux.;
4. gcc для хост-машины.;
5. Бинарные утилиты или binutils представляют собой набор инструментов программирования для манипулирования форматами объектных файлов. Обычно они используются вместе с коллекцией компиляторов GNU и GDB;
6. Любой программный стек, который мы можем использовать.

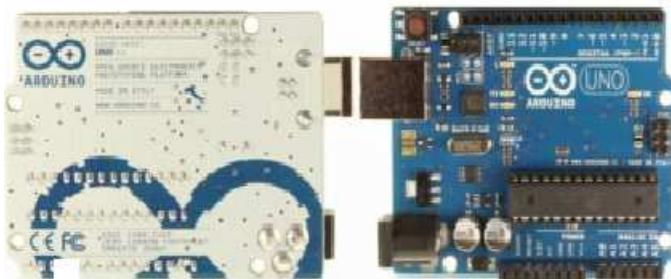


Рисунок 1 – Плата Arduino UNO R3

Операционная система Linux в сочетании с мощностью пульта управления предоставляет нам интеллектуальную базу, которая может быть настроена в соответствии с требованиями. UGV был создан с использованием 8-битного микроконтроллера, который дополняет функциональность используемой нами платы ARM. Камера фиксирует изображения, и они беспроводным образом передаются от платы ARM к Хосту. Плата ARM получает сигнал от главного компьютера. через беспроводной ключ. 8-битный микроконтроллер управляет беспроводным пультом дистанционного управления игрушечной машинкой. Использование базовой настройки

Взяв за основу любую простую игрушечную машинку, вертолет или корабль, которые можно использовать для отправки прямой трансляции. Устройство может затем можно управлять удаленно и добавлять персонализированные функции. Дистанционно управляемый UGV может использоваться в целях обороны, добычи полезных ископаемых и других мест, куда отправляются люди могут быть опасны при низких затратах. UGV можно заменить, а человеческую жизнь - нет.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Зильбершатц, Гальвин и Ганье, Принципы операционной системы.
2. Вольфганг Мауэрер, Архитектура ядра Linux: Wiley India.
3. Корбет, Рубини, Кроа-Хартман, Драйверы ядра Linux: О'Рейли.
4. Griffon: переносной гибридный UGV / UAV », Брайан Ямаути и Павел Рудакевич, Промышленный робот, т. 31, нет. 5. С. 443-450, 2004.
5. Полуавтономная беспилотная система управления наземным транспортным средством. Джонатан Андерсона, Да-Дже Ли, Роберт Шенбергер, Чжаойи Вейя и Джеймс Арчибальда.

УДК 004.021

В.О. ЗИНОВА, А.Б. ЧЕРЕНКОВАРязанский государственный радиотехнический университет
имени В.Ф. Уткина**РАЗРАБОТКА МОДИФИЦИРОВАННОГО ГЕНЕТИЧЕСКОГО
АЛГОРИТМА ДЛЯ КОМПОНОВКИ БЛОКОВ ЭЛЕКТРОННО-
ВЫЧИСЛИТЕЛЬНЫХ СРЕДСТВ И ЕГО ПРИМЕНЕНИЕ ДЛЯ
АДАПТАЦИИ ДАННЫХ**

Рассматривается разработка генетических алгоритмов применительно к блокам электронно-вычислительных средств, а также возможности таких алгоритмов касемо адаптации и генерации данных.

Генетический алгоритм - это эвристический метод, точнее, один из видов эволюционных алгоритмов, использующий идею и терминологию эволюции природы. Генетический алгоритм отлично показывает себя при решении нестандартных задач, проблем с неполными данными или при невозможности использования других методов оптимизации из-за сложности исполнения или продолжительности исполнения. Генетический алгоритм считается завершенным, если пройдено определенное количество итераций или если было получено удовлетворительное значение функции приспособленности. Как правило, генетический алгоритм решает проблему максимизации или минимизации, а адекватность каждого решения оценивается с помощью функции приспособленности. Генетический алгоритм позволяет эффективно учитывать множество входных ограничений, обладая при этом малой временной сложностью. Именно поэтому он является эффективным средством для решения задач компоновки блоков ЭВС большой размерности.

Решение задачи компоновки блоков электронно-вычислительных средств основывается на их коммутационной схеме. Для алгоритмизации и формального решения задачи производится переход от коммутационной схемы к графу или гиперграфу [2]. Исходя из этого, решение задачи сводится к разбиению графа на заданное количество частей, что в свою очередь является NP-полной и NP-сложной задачей.

Разбитие графа на части как правило может производиться несколькими методами. При тривиальном методе разбиения с использованием полного перебора находятся все возможные варианты разбиения, а затем из них отбирается наиболее подходящий.

Количество вариантов компоновки элементов в блоки находится по следующей формуле:

$$\tilde{C}_n^m = \frac{(n+m-1)!}{m!(n-1)!}. \quad (1)$$

Здесь n – количество блоков; m – количество элементов.

При увеличении числа элементов или блоков электронно-вычислительного средства количество возможных вариантов их сочетания резко увеличивается и на их компоновку с использованием метода полного перебора потребуется предельно большое количество машинного времени. Таким образом, разработка нового метода компоновки блоков электронно-вычислительных средств, нетребовательного к машинным и временным ресурсам, является актуальной задачей.

Для применения генетического алгоритма к задаче компоновки его необходимо модифицировать. Суть самого алгоритма изменяться не должна. Модифицированный генетический алгоритм начинает работу при загрузке матрицы инцидентности. Вывод лучшей особи свидетельствует об окончании работы.

Дадим словесное описание схемы модифицированного алгоритма:

1. Формирование начальной популяции. На этом этапе создается новая особь. Также необходимо повторно создать определенное количество особей, чтобы размер популяции был равен исходному значению размера популяции.

2. Вычисление полезности каждой особи.

3. Выбор очередной пары родителей.

4. Скрещивание выбранной пары особей. Итог- создание новой особи, содержащей гены обоих «родителей».

5. Расчет полезности (ценности) новой особи. Добавление потомка в текущую популяцию. После вычисления полезности новой особи, необходимо добавить ее в текущую популяцию.

6. Циклический процесс скрещивания. Продолжается до тех пор, пока все пары особей не будут скрещены между собой. Выбирается наиболее ценный потомок.

7. Мутация особей в популяции. Процесс мутации заключается в изменении случайного гена. Если произошло переполнение блока, то в нем выбирается случайный элемент (кроме текущего) и он переходит в предыдущий блок.

8. Вычисление полезности новых особей. После процесса мутации вновь вычисляется полезность каждой особи и выбирается наилучшая.

9. Отбор наиболее полезных особей методом усечения. В нем производится быстрая сортировка. Особи, которые выбыли, автоматически погибают (исключаются из популяции).

10. Условием остановки работы генетического алгоритма является количество поколений, потому что в задаче компоновки не существует локальных экстремумов – после операций скрещивания и мутации может оказаться особь, которая станет новой «лучшей» особью в популяции. В том случае если предельное число поколений не достигнуто, продолжают выше описанные процессы с оставшимися после отбора особями. Итог – получение наиболее эффективного решения. Выводится самая приспособленная особь

Использование генетических алгоритмов в процессе адаптации данных позволяет гарантировать, что мы определим наиболее сложные части программного кода, в которых риски из-за ошибок наиболее велики [1]. Оценка выполняется с помощью фитнес-функции, параметры которой имеют различный вес каждой отдельной операции. На сегодняшний день разработано множество типов диаграмм, которые позволяют представить структуру программного кода не как набор действий, а как диаграммы с определенной структурой. Наиболее широко используемые представляют собой диаграммы (графики) потоков управления, которые позволяют представить все многообразие способов запуска программы. Основное назначение таких диаграмм - создание программного кода, который включает определение сложности программы, проверку логики программы и непосредственное написание кода. Однако с точки зрения проблемы генерации тестовых данных этот тип диаграмм, построенных на уже написанном программном коде, позволяет оценить качество разработанного кода и, в рамках задачи, оценить важность или сложность некоторых программных путей.

На основе возможности представления программного кода в структурной форме был разработан подход, с помощью которого можно было бы оценить программный код и определить такой набор тестовых данных, позволяющий «пройти» наибольшее количество операций и наибольшее количество путей. Первым шагом в предлагаемом методе является рассмотрение структурных элементов кода. Для простоты представления мы можем использовать блок-схемы для визуализации структуры кода и понимания того, как выполняется программа. Каждой операции кода назначается отдельный узел графа, а в качестве ссылки указывается направление, в котором выполняется код.

Задача генерации входных тестовых данных состоит из трех подзадач: поиск входных данных для обхода одной из самых сложных ветвей кода; устранение или уменьшение весов операций, лежащих на этой ветви кода, со скоростью последующих ветвей; поиск набора тестовых данных для одновременного прохождения нескольких ветвей. Ограничение на размер входного набора данных устанавливается после этапа разработки и позволяет сконцентрироваться на определенных отраслях, в которых выполняется наибольшее количество операций. Весь алгоритм выполняется циклически. Сначала запускается процедура поиска входных данных для одной ветви, затем операции в этой ветви исключаются из дальнейших вычислений, и снова запускается поиск данных для одной ветви.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Вороновский Г.К., и др. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности / – Х.: Основа, 1997. – 112 с.
2. Курейчик В.М. Математическое обеспечение конструкторского и технологического проектирования с применением САПР: учебник для вузов / В. М. Курейчик. – М.: Радио и связь, 1990. –352 с.

УДК 004.048

М.А. ИВАНЧИКОВА, Д.С. ТУРБИНА
Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

АНАЛИЗ ДАННЫХ ДЛЯ ПРОГНОЗИРОВАНИЯ ПОТОКА АБИТУРИЕНТОВ ПО НАПРАВЛЕНИЯМ ПОДГОТОВКИ ВЫСШЕГО ОБРАЗОВАНИЯ

Приведены результаты построения тепловой карты корреляции исходного набора данных и анализа соотношения количества бюджетных мест и поданных заявлений на направления подготовки.

Прогнозирование потока абитуриентов является актуальной задачей для каждого ВУЗа. В последнее время условия учебной и финансовой деятельности являются метриками оценки уровня университета у абитуриентов. Поэтому модель, позволяющая сделать выводы о динамике изменения численности абитуриентов, основанные на комбинациях влияющих характеристик, поможет выявить сильные и слабые стороны приемной кампании и принять решения по ее улучшению.

На первом этапе исследования был проведен сбор первоначальных данных, их очистка и преобразование. Была сформирована таблица с основными выявленными признаками, включающими контрольные цифры приема и данные абитуриентов [1].

Для анализа полученного набора данных была проведена визуализация и определения корреляции признаков с помощью методов машинного обучения, реализованных с помощью языка программирования Python. На рисунке 1 показана полученная тепловая карта корреляции.

Анализируя полученную карту, можно наблюдать максимальную взаимосвязь (значения от 0,5 до 1 и от -0,5 до -1) для пар «Количество бюджетных мест – Количество заявлений на бюджетные места» (0,99), «Количество бюджетных мест – Сумма баллов по вступительным испытаниям» (0,53), «Сумма баллов по вступительным испытаниям – Сумма баллов» (0,89), «Сумма баллов по вступительным испытаниям – Балл аттестата» (0,71), «Сумма баллов по вступительным испытаниям – Место проживания» (0,5), «Сумма баллов по вступительным испытаниям – Направление» (0,6), «ЕГЭ по физике – ЕГЭ по математике» (0,57), «ЕГЭ по физике –

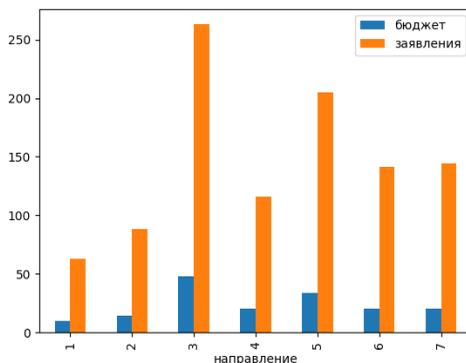


Рисунок 3 – Соотношение количества бюджетных мест и поданных заявлений

Для оценки уровня подготовки студентов также важное значение имеет средний балл аттестата. В основном абитуриенты имеют балл от 4,0 до 5,0. Кроме того, для выбора направления важна сумма баллов за ЕГЭ или вступительные испытания. Количество баллов, учитывающихся в конкурсе, состоит не только из баллов по ЕГЭ или вступительным испытаниям, а также включает в себя дополнительные баллы за золотую медаль, участие в олимпиадах и др. По значениям корреляции мы можем наблюдать сильную взаимосвязь между наличием золотой медали и дополнительными баллами за олимпиады (0,83). Поэтому можно сделать вывод, что абитуриент, являясь в школе отличником, также проявлял себя и на олимпиадах, и скорее всего, имеет углубленные знания.

Очевидной является связь баллов по отдельным предметам ЕГЭ с общей суммой баллов. Но интересно заметить то, что сильную взаимосвязь имеет пара «ЕГЭ по математике – ЕГЭ по физике». Отсюда мы можно сделать вывод, что понимание физики абитуриентом связано с пониманием математики – это то, что и требуется от студента технического ВУЗа.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Иванчикова М.А., Турбина Д.С. Предобработка данных для прогнозирования потока абитуриентов по направлениям подготовки высшего образования // Сборник трудов XXV всероссийской научно-технической конференции студентов, молодых ученых и специалистов Новые информационные технологии в научных исследованиях и образовании: «НИТ-2020»

УДК 004.9

М.С. ИГОШИНА, В.В. ЯШКОВА.

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

СРАВНИТЕЛЬНЫЙ АНАЛИЗ АЛГОРИТМОВ РАЗМЕЩЕНИЯ

В статье проводится краткий сравнительный анализ некоторых алгоритмов размещения, рассматривается их классификация, описывается суть методов, достоинства и недостатки, приводятся примеры области применения.

В настоящее время при разработке вычислительных комплексов используют модульный принцип, который базируется на построении ЭС из модулей, разбитых на несколько уровней иерархии. Одной из ключевых конструкторских задач, типовой для всех модулей является задача размещения.

Размещение – определение местоположения элементов в заданном монтажном пространстве, удовлетворяющее некоторым требованиям. Такая задача носит многоцелевой характер, а при её решении происходит оптимизация по совокупности критериев.

Все алгоритмы размещения можно разбить на следующие большие группы:

- Математические модели;
- Конструктивные алгоритмы начального размещения;
- Итерационные алгоритмы размещения;
- Непрерывно-дискретные методы.

В данной статье мы проанализируем некоторые методы, входящие в эти классы.

К математическим моделям относятся: модель квадратичного назначения, метод ветвей и границ, а также непрерывные модели.

Задача квадратичного назначения есть задача размещения по критерию минимизации суммарной взвешенной длины соединений, которая сводится к минимизации функционала на множестве перестановок p . Набор позиций считается фиксированным, элементы рассматриваются, как геометрические точки, схемы соединений исходят из центров элементов, кроме того, центры элементов и позиций совпадают. Для решения используют алгоритмы, основанные на методе ветвей и границ. Суммарная взвешенная длина при произвольном размещении будет рассчитываться следующим образом:

$$F(p) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n r_{ij} d_{p(i)p(j)} + \sum_{i=1}^n a_{ip(i)},$$

где $r_{ij} d_{p(i)p(j)}$ – длина соединений между элементами e_i и e_j .

Достоинства: возможность получения точного решения при $n=15 \dots 20$.

Недостатки: значительные временные затраты на поиск решения при большом количестве элементов (n).

Особенностью непрерывных моделей является размещение элементов на непрерывной плоскости, когда их положения заранее не определены. В данном случае процесс размещения включает 2 шага:

- Решение непрерывной задачи;
- Преобразование полученного решения к дискретному множеству позиций.

Решение сводится к поиску минимума, через систему алгебраических уравнений. На этом этапе выявляется недостаток модели – при размещении элементы располагаются в одной точке. Для компенсации такого явления используют предварительную фиксацию некоторых элементов (например, внешних выводов).

Достоинства: наличие стандартных программ для решения задач, а также малые временные затраты.

Недостатки: возможность перекрытия конфигурации элементов, неравномерность распределения элементов, нахождение только локального экстремума.

Конструктивные алгоритмы начального размещения включают в себя 2 подгруппы – последовательные и параллельно-последовательные алгоритмы. Отличаются значительным быстродействием.

Матричные методы размещения относятся к последовательным алгоритмам, т.е. процесс назначения элементов последователен. Их суть заключается в вычислении на каждой итерации приращения целевой функции, а именно – суммарной длины соединений каждого неразмещенного элемента в каждую из незанятых позиций, что позволяет создать матрицу:

$$A = \|a_{ij}\|_{(n-k+1) \times (n-k+1)},$$

где a_{ij} – цена назначения элемента $e_i \in \bar{E}_k$ в позицию $l_j \in \bar{L}_k$.

Существуют разные способы выбора «цены назначения» такие как, принцип максимина, линейное назначение, принцип минимального риска, принцип минимальных потерь. Из всех

перечисленных, алгоритм минимальных потерь позволяет достичь меньших значений суммарной длины соединений.

Достоинства: быстроедействие.

Недостатки: невысокая точность, которая ухудшается с увеличением количества элементов.

Метод разбиения является параллельно-последовательным алгоритмом размещения. Он состоит из 2-х шагов и сводится к разбиению схемы на множества с уточнением позиции этих множеств на контактной площадке. На каждой итерации очередное множество разбивается, с помощью эвристического алгоритма, на два равных подмножества элементов. Параллельно с этим производится ограничение позиций элементов в пространстве, занятом первоначальным множеством.

Достоинства: малое машинное время работы, равномерное распределение элементов по всей поверхности КП.

Недостатки: узкая область применения.

Итерационные алгоритмы подразделяются на алгоритм парных перестановок, алгоритм групповых перестановок, методы случайного поиска. Для них характерно наличие начального размещения, для обнаружения в нем нового размещения (с меньшим критерием), которое принимается за начальное. Итерации повторяются до тех пор, пока при размещении не будут отсутствовать варианты с меньшим значением критерия. Позволяют достичь более точные результаты.

Метод случайного поиска сводится к следующему: задается число вариантов распределения элементов, после этого элементы распределяются в позиции (равномерное или экспоненциальное распределение) согласно выбранному критерию оптимизации, лучший результат есть искомое решение.

Достоинства: подходят для таких задач, как получение плотных укладок разногабаритных элементов.

Недостатки: малоэффективны по сравнению с детерминированными методами.

Непрерывно-дискретные методы включают в себя метод силовых функций и метод последовательного сдвига. Лучше всего подходят для размещения разногабаритных элементов.

В основе метода силовых функций лежат непрерывные модели, описанные ранее. Суть алгоритма заключается в следующем: элементы рассматриваются, как некие материальные точки, к которым применены силы – силы отталкивания и притяжения. Сила отталкивания определяется как:

$$F_{ij} = r_{ij} f(x_i, y_i, x_j, y_j),$$

где $|f(x_i, y_i, x_j, y_j)| \leq r_f d_{ij}$, r_f – число соединений между элементами.

Здесь сила отталкивания позволяет достичь равномерного расположения элементов, без перекрытия, а сила притяжения определяет расстояние между ними. Сила притяжения вычисляется как:

$$\Phi_{ij} = \varphi_{ij}(x_i, y_i, x_j, y_j),$$

где $|\varphi(x_i, y_i, x_j, y_j)| \leq r_\varphi d_{ij}$, r_φ – коэффициент.

После чего производится размещение элементов на плоскости, т.е. нахождение состояния равновесия системы.

В динамическом варианте данного метода для определения состояния равновесия составляется система дифференциальных уравнений, в случае статического варианта данного метода – уравнения статического равновесия, для которых необходимо, чтобы результирующие силы, действующие на каждую точку, равнялись 0. Ещё одна вариация метода, позволяет рассматривать задачу размещения с помощью линейных схем.

Достоинства: возможность получения глобального экстремума целевой функции, а также наличие стандартных программ пригодных для решения задач данного класса.

Недостатки: значительные и трудоемкие вычислительные затраты, особенно это проявляется при большом числе элементов.

Метод последовательного сдвига компенсирует этот недостаток за счёт аналитического определения позиции элемента, вместо перебора приращений целевой функции.

Такой алгоритм есть модификация метода силовых функций. Он базируется на последовательном сдвиге каждого элемента в такую позицию, результирующая сила, действующая на него со стороны других элементов, будет минимальна. Критерием оптимальности выступает оценка длины много концевых соединений, рассчитываемая как:

$$L = \sum_{j=1}^M l(v_j),$$

где v_j – элементный комплекс,

$$l(v_j) = |x_{jmax} - x_{jmin}| + |y_{jmax} - y_{jmin}|,$$

где $x_{jmax}, x_{jmin}, y_{jmax}, y_{jmin}$ – максимальные и минимальные координаты элементов, входящие в элементное множество.

Одна итерация включает размещение всех элементов $i=1, 2, \dots, n$, после чего последовательность действий повторяется. Число итераций определяется при экспериментальном изучении алгоритма.

Достоинства: монотонное уменьшение целевой функции за счет локальной оптимизации положений отдельных элементов, следовательно, значительное сокращение времени на достижение требуемого результата.

Недостаток: возможность образования «стягивания» элементов в отдельные центры группирования.

Таким образом был проведен краткий сравнительный анализ алгоритмов размещения. При выборе метода решения задачи следует опираться на такие факторы, как размерность задачи, структура схем соединения и т.д. От них будет зависеть эффективность отобранного для использования алгоритма, ведь наиболее оптимальный способ для одного класса задач, не будет таковым являться для другого класса.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Абрайтис Л.Б., Жилевичюс В.А. Исследование эффективности алгоритмов размещения. - В кн.: Вычислительная техника. Т.2, Каунас, 1971, с.86-93.

2. Абрайтис Л.Б., Лазаускас Д.А. Простой алгоритм размещения. В кн.: Вычислительная техника. Т.6, Каунас, 1974, с.83-86.

3. Бахтин Б.И. Статистический метод оценки эффективности алгоритмов размещения компонентов электронной аппаратуры.- В кн.: Вычислительная техника. Т.1, Каунас, 1970, с.282-287.

4. Белов А.Ф., Горячев А.А., Курков Е.В. Обзор алгоритмов размещения компонентов на печатных платах. Ядерное приборостроение, 1972, вып.18, с.22-31.

5. Белов А.Ф., Курков Е.В., Горячев А.А., Зайцева Н.П. Решение задачи размещения модулей на плате по минимуму суммарной длины связей на ЭВМ "Минск-32". Ядерное приборостроение, 1972, вып.13, с.40-47.

6. Бугаев Е.С., Юрин О.И. Общий алгоритм размещения элементов с нефиксированными связями. В кн.: Автоматизированные системы проектирования. М., МЛНТП, 1975, с.90-94.

7. Сапрыкин А.Н. Конспект лекций по курсу «Информационные технологии проектирования ЭВС».

УДК 004.021

Т.А КАЛИНИН, М.А. ПАНКОВАРязанский государственный радиотехнический университет
имени В.Ф. Уткина**СРАВНЕНИЕ АЛГОРИТМОВ ПОСТРОЕНИЯ ДЕРЕВЬЕВ
ШТЕЙНЕРА В ОРТОГОНАЛЬНОЙ МЕТРИКЕ**

В статье сравниваются алгоритм линейного поиска и алгоритм описывающих прямоугольников. Критерием оценки будет выступать длина итогового связывающего дерева.

Введение.

Задача Штейнера формулируется следующим образом:

На плоскости задано n точек. Требуется соединить эти точки ломаными линиями так, чтобы каждая точка была соединена с каждой, и чтобы суммарная длина всех проведенных линий была минимальна [1]. Дополнительные вершины, по-другому точки Штейнера, устанавливаются в точках пересечения ломаных линий.

Деревья Штейнера имеют широкую область применения и используются при разработке компьютерных микросхем, междугородных телефонных линий, многоадресной маршрутизации в сетях телекоммуникаций и т.д.

Здесь мы рассмотрим применение деревьев Штейнера для трассировки печатных плат. Построение деревьев мы будем производить в ортогональной метрике, поскольку при машинной трассировке печатных соединений, как правило, используется ортогональная опорная сетка.

Для построения деревьев Штейнера мы будем использовать алгоритм линейного поиска и алгоритм описывающих прямоугольников. Рассмотрим эти алгоритмы подробнее.

Алгоритм линейного поиска.

Данный алгоритм является приближенным методом построения дерева Штейнера в ортогональной метрике.

Пусть $P = \{p_i\}$ – множество исходных точек. Алгоритм построения дерева [2] включает следующие операции:

- 1) Пронумеровать точки множества P по спирали (по часовой стрелке);
- 2) Присвоить каждой точке p_i один из двух признаков, определяющих направление преимущественного проведения трассы от этой точки: 0 – вдоль оси X , 1 – вдоль оси Y .

Признаки 0 и 1 присваиваются попеременно, начиная с точки $p_1 = p_i$, которая помечается признаком 0. При этом точки, находящиеся на одной горизонтали (вертикали) получают один и тот же признак, признак присваивается только 1 раз;

3) Начиная с точки p_1 , найти точку p_i , для которой $d_{i1} = |x_i - x_j| + |y_i - y_j| = \min d_{j1}$. Построить ребро (фрагмент дерева), соединяющий p_1 с p_i , выбирая из двух возможных вариантов начального направления ребра тот, который соответствует признаку точки p_i ;

4) Всем вершинам сетки, через которые прошел фрагмент, присвоить наименьший из номеров концевых точек фрагмента;

5) Шаг 3 и 4 выполнять для остальных точек в порядке возрастания номеров до тех пор, пока не будут построены фрагменты для всех точек p_i ;

6) Повторять шаги 3-5 до тех пор, пока все p_i не получат номер 1.

Алгоритм описывающих прямоугольников.

Данный алгоритм применяется для построения квазиоптимального дерева в ортогональной метрике. Преимуществом данного метода, помимо минимизации дерева, также является то, что он нацелен на минимизацию количества точек Штейнера.

Пусть $P = \{p_i\}$ – множество исходных точек, $\Pi = \{\Pi_i\}$ – множество описывающих прямоугольников, а $V = \{V_i\}$ – множество периметров, принадлежащих Π_i прямоугольникам.

Алгоритм построения дерева [3] включает следующие операции:

1) Строим систему описывающих прямоугольников $\{\Pi_i\}$. Прямоугольник Π_1 строится минимальным по площади со сторонами, параллельными образующим ортогональной сетки, и содержащим внутри себя множество вершин $\{p_i\}$. Некоторые из них будут лежать на периметре V_1 прямоугольника Π_1 .

Прямоугольник Π_i строится аналогично прямоугольнику Π_1 , только на множестве вершин $N \setminus V_1$ и т. д. Последний прямоугольник Π_k может вырождаться в точку или в отрезок, либо остаться прямоугольником;

2) Выбираем точку на периметре V_1 прямоугольника Π_1 и считаем расстояния: d_1 – расстояние до ближайшей точки на периметре первого прямоугольника; d_2 – расстояние до границы следующего прямоугольника (считаем, что $V_2 \neq \emptyset$);

3) Сравниваем d_1 и d_2 .

Если $d_1 > d_2$, то вершина p_1 соединяется с ближайшим узлом на V_2 . Если в этом узле нет вершины, принадлежащей P , то фиксируется дополнительная вершина.

Если $d_1 \leq d_2$ то p_1 соединяется с ближайшей вершиной на V_1 . Предположим, что это вершина p_2 . Если вершины p_1 и p_2 не лежат на одной прямой, то они соединяются двумя отрезками, обращенными углом к следующему (меньшему) прямоугольнику, на точку пересечения отрезков фиксируется дополнительная вершина; если обе вершины лежат на одной прямой образующей сети, то они соединяются отрезком прямой;

4) Повторяем шаги 2 и 3 для точек, принадлежащих периметру V_1 ;

5) Если все точки на периметре V_1 соединены с периметром V_2 непосредственно или через другие вершины, переходим к следующему прямоугольнику P_2 , на котором соединяем как основные, так и появившиеся дополнительные вершины (шаги 2 – 4). Если все прямоугольники были пройдены, переходим к следующему шагу;

6) Производим анализ дополнительных вершин.

Берется произвольная вершина со степенью 1, пусть это будет p_1 .

Если вершина p_1 соединена ребром с основной вершиной или вершиной со степенью более 2, то исключаем ее из рассмотрения и переходим к следующей вершине.

Если вершина p_1 соединена с дополнительной вершиной со степенью 2, то считаем длину пути d_{p_1} от вершины p_1 до первой встретившейся основной вершины, либо до дополнительной вершины со степенью более 2, двигаясь при этом по ребрам построенного дерева. Сравниваем длину пути с расстоянием d'_{p_1} от p_1 до ближайшей вершины, лежащей на том же прямоугольнике, что и p_1 . Если $d_{p_1} > d'_{p_1}$, то в дереве путь, соответствующий d_{p_1} , стирается, и строится ребро, соответствующее расстоянию d'_{p_1} . Заметим, что связность дерева при этом не нарушается;

7) Выравнивание углов.

Если дополнительная вершина имеет общее ребро с другой дополнительной вершиной, то она переводится в диагонально противоположный угол прямоугольника, построенного на двух соседних вершинах.

Если дополнительная вершина имеет степень более 2, то в диагонально противоположный угол переводится соседняя дополнительная вершина. Нетрудно заметить, что построенное таким образом дерево полностью лежит на образующих, проходящих через основные точки-вершины.

Контрольный пример.

Для наглядности рассчитаем пример для обоих алгоритмов и сравним полученные деревья.

Набор координат точек будет одинаков для обоих алгоритмов. Координаты: $x_1(0;5)$, $x_2(0;7)$, $x_3(2;7)$, $x_4(6;2)$, $x_5(8;5)$, $x_6(5;10)$, $x_7(9;11)$, $x_8(12;7)$, $x_9(9;14)$, $x_{10}(13;14)$.

Для начала рассчитаем пример для алгоритма линейного поиска.

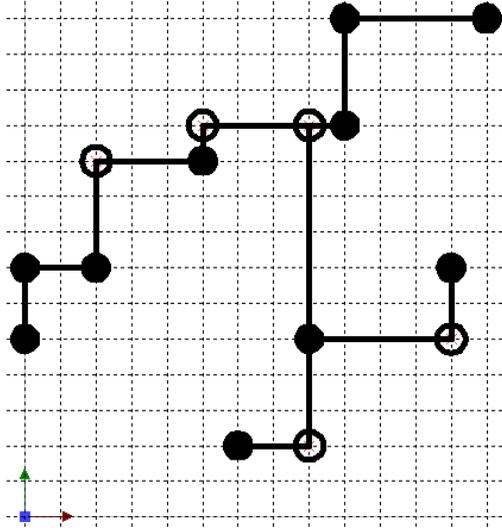


Рисунок 1 – Алгоритм линейного поиска

Длина полученного дерева – 39.

Количество точек Штейнера – 5.

Далее рассчитаем пример для алгоритма описывающих прямоугольников.

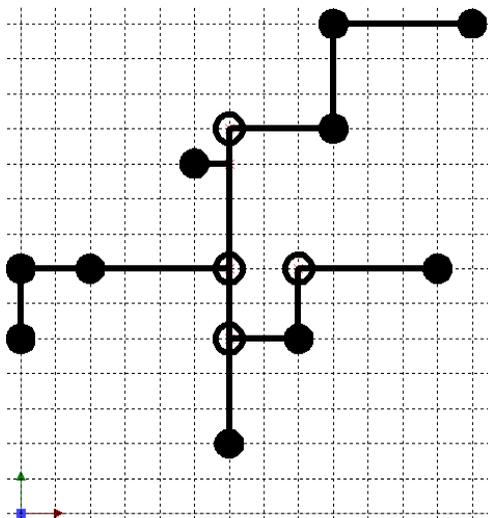


Рисунок 2 – Алгоритм описывающих прямоугольников

Длина полученного дерева – 37.

Количество точек Штейнера – 4.

Сделаем вывод по полученным результатам.

- 1) Структура деревьев сильно различается;
- 2) Длина дерева построенного по алгоритму линейного поиска больше, чем длина дерева, построенного по алгоритму описывающих прямоугольников. Из этого можно сделать первичный вывод, что дерево, построенное по алгоритму описывающих прямоугольников, эффективнее;

3) Также можно отметить, что количество точек Штейнера во втором дереве меньше, чем в первом, что также говорит об эффективности метода описывающих прямоугольников.

Программный расчет.

Далее произведём сравнение с помощью программного расчета.

С помощью программы построим 100 деревьев для 10, 100 и 1000 вершин, а также вычислим их средние длины в категории. Сведем результаты программного расчета в таблицу и сравним полученные результаты.

Условия проведения расчетов:

- Координаты вершин будут заданы с помощью генератора случайных чисел;

- Деревья будут строиться по одним и тем же координатам для обоих алгоритмов;
- Поле, на котором будут строиться деревья, ограничено квадратом 100×100 .

Таблица 1.

Количество деревьев	Линейный поиск	Описывающие прямоугольники
10	230	211
100	773	705
1000	2398	2203

Вывод.

По результатам произведенных расчетов можно сделать вывод, что алгоритм описывающих прямоугольников является более эффективным, чем алгоритм линейного поиска. По главному критерию - результирующие деревья получаются более короткими, также можно учесть, что в дереве используется меньше точек Штейнера.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. И.В. Романовский. Задача Штейнера на графах и динамическое программирование. Наши авторы 2004 г. 7 с.
2. В.А. Селюгин. Машинное конструирование электронных устройств. М., «Сов. радио», 1997, 384 с.
3. Г.А. Петухов, Г.Г. Смолич, Б.И. Юлин. Алгоритмические методы конструкторского проектирования узлов с печатным монтажом. М., «Радио и связь», 1987, 152 с.

УДК 004.65

А.Н. КАЩЕЕВРязанский государственный радиотехнический университет
имени В.Ф. Уткина**ВЫЧИСЛЕНИЕ МАТЕМАТИЧЕСКИХ ФУНКЦИЙ
НА ЯЗЫКЕ C++**

В данной работе будет рассмотрен один из способов вычисления математических функций при помощи алгоритма написанного на языке высокого уровня C++. В статье будет расписан пример вычисления натурального логарифма $\ln(x)$ для любого допустимого числа x .

Известно, что для аппроксимации математических функций используется ряд Тейлора:

$$f(x) = \sum_{k=0}^N \frac{f^{(k)}(x_0)}{k!} (x - x_0)^k + \theta(x, x_0); \quad (1)$$

Где $f(x)$ – функция, которую требуется разложить; $\theta(x, x_0)$ – остаточный член в форме Пиано; x – переменная; x_0 – «точка» относительно которой будет происходить аппроксимация. Если $N \rightarrow \infty$ то $\theta(x, x_0) \rightarrow 0$.

Пусть мы хотим разложить в ряд Тейлора натуральный логарифм $\ln(x)$. Перед тем как применить формулу (1) выберем точку начала аппроксимации x_0 . Так как $\ln(0) = -\infty$, то целесообразно принять $x_0 = 1$. Теперь применим формулу (1) и проведя математические преобразования получим:

$$f(x) = \ln(x) = \sum_{k=1}^N \frac{(-1)^{k+1} \cdot (x-1)^k}{k}. \quad (2)$$

Полученное выражение не является конечным результатом, так как эта формула справедлива не для всех x . Чтобы в этом убедиться построим график для $N=5$ (см. рисунок 1). Из графика видно, что полученная функция $f_1(x)$ совпадает с желаемой лишь на коротком интервале (в районе где $x = 1$). При больших значениях N функция $f_2(x)$ все равно не совпадает с желаемой:

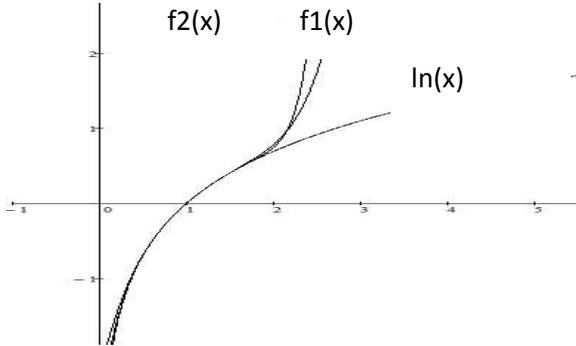


Рисунок 1 – сравнение графиков

Из этого следует, что не все функции разлагаются в ряд Тейлора (сходимость ряда). Чтобы исследовать на сходимость ряда воспользуемся признаком Даламбера:

$$\lim_{k \rightarrow \infty} \left| \frac{f_{k+1}(x)}{f_k(x)} \right| = l(x) \quad (3)$$

При $l(x) > 1$ – ряд расходится; при $l(x) < 1$ – ряд сходится; при $l(x) = 1$ – неопределенность.

Тогда для нашего случая используя формулу (3) получим:

$$\lim_{k \rightarrow \infty} \left| \frac{(-1)^{k+2} \cdot (x-1)^{k+1}}{(-1)^{k+1} \cdot (x-1)^k} \cdot \frac{k+1}{k} \right| = |x-1| \lim_{k \rightarrow \infty} \left| -\frac{k}{k+1} \right| = |x-1|.$$

Для того чтобы ряд сходилась надо чтобы выполнялось условие:

$$0 < x < 2 \quad (4)$$

То есть формула (2) справедлива для $x \in (0; 2)$ где $x \in \mathbb{R}$, что и видно на графике выше (рисунок 1).

Для того чтобы вычислить натуральный логарифм для $x > 0$ воспользуемся формулой:

$$\ln(x) = \ln(a \cdot b) = \ln(a) + \ln(b) \quad (5)$$

Где a – остаток от деления на экспоненту, а $b = \exp(n)$; $n \in \mathbb{N}$ – кратность числа e , т.е. сколько раз надо поделить на экспоненту чтобы выполнялось условие (4).

Например:

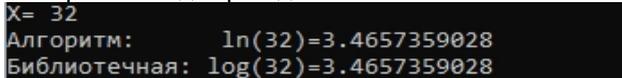
$$\ln(9) = \ln\left(\frac{9}{e^2} \cdot e^2\right) = \ln\left(\frac{9}{e^2}\right) + \ln(e^2) = \ln\left(\frac{9}{e^2}\right) + 2$$

Здесь для первого слагаемого применима формула (2).

Реализация алгоритма на языке C++ представлена ниже:

```
#include <iostream>
#include <iomanip>
#include <math.h>
using namespace std;
double step (double x, int y) {
    double z=1;
    for (int i=0; i<y;i++)
        z=z*x;
    return z;
}
double ln (double x) {
    int n=0;
    double result=0;
    while (x>2){
        n++;
        x=x / 2.718281828459;
    };
    for (int k=1; k<100; k++)
        result=result+(step(-1, k+1) *step(x-1, k)) /k;
    return result+n;
}
int main (){
    setlocale(0, "");
    double x;
    cout <<"X= ";
    cin>>x;
    if (x<=0) return 0;
    cout<<setprecision(13);
    cout <<"Алгоритм: ln ("<<x<<")="<<ln(x) <<"\n";
    cout <<"Библиотечная: log ("<<x<<")="<<log(x);
    return 0;
}
```

Результат работы кода приведен ниже:



```
X= 32
Алгоритм:      ln(32)=3.4657359028
Библиотечная: log(32)=3.4657359028
```

Рисунок 2 – Результат выполнения программы

Таким образом при помощи разложения в ряд Тейлора можно вычислять многие математические функции, но для этого необходимо разложить функцию в ряд Тейлора и исследовать ее на сходимость. При случаях, когда ряд частично сходится следует применять математические «приемы» для выполнения условия сходимости.

УДК 004.032.26

П.С. КИРЕЕВ, С.В. СКВОРЦОВ

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

ПРИМЕНЕНИЕ СВЁРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ РЕШЕНИЯ ЗАДАЧИ РАСПОЗНАВАНИЯ ИЗОБРАЖЕНИЙ

Рассматриваются принципы работы нейронных сетей со свёрточной архитектурой на примере модели VGG16 и их применение для организации автоматизированного сбора данных об объектах по фотографиям в информационных системах.

Задача машинного распознавания изображений в настоящее время является одной из наиболее востребованных в различных областях науки и техники. Основная проблема заключается в том, что в связи с различными индивидуальными особенностями изображений, получить удовлетворительное решение данной задачи алгоритмическими способами не представляется возможным. Поэтому данный класс задач относится к области искусственного интеллекта, в которой основными моделями являются нейронные сети, а вычислительные алгоритмы заменяются процедурой обучения этих сетей.

Одной из областей, для которых актуальна задача распознавания изображений, является разработка автоматизированных информационных систем (АИС), хранящих помимо прочих типов графические данные (фотографии объектов). Разработка модуля, организующего автоматизированный сбор информации об объектах по их фотографиям, значительно ускорит работу операторов таких АИС.

Для решения задачи распознавания изображений были разработаны специальные модели нейронных сетей со свёрточной архитектурой (Convolutional Neural Network или сокращённо CNN). За идею была взята работа Дэвида Хьюбеля и Торстена Визеля, в которой они исследовали устройство части мозга кошки, отвечающей за визуальное восприятие [1].

На вход CNN подаётся изображение в виде набора матриц, каждая из которых характеризуют интенсивности пикселей своего цвета. Такие матрицы называются каналами, а их совокупность - картой признаков. Количество каналов входного слоя зависит от цветовой модели изображения. CNN состоит из свёрточных и полносвязных слоёв.

Свёрточный слой представляет из себя набор обучаемых фильтров, каждый из которых отвечает за распознавание какого-либо признака на изображении. Применение этих фильтров ко всем элементам каналов, поступающим на вход свёрточного слоя, позволяет преобразовать их в новую карту признаков. Таким образом, последующие свёрточные слои пытаются распознать всё более сложные элементы путём комбинации простых признаков, поступающих к ним на вход из предыдущих слоёв. Математически результат применения фильтра свёрточного слоя к элементу входного канала можно записать следующим образом:

$$O_{xyc'} = \sum_{i,j,c} W_{ijcc'} X_{x-iy-j,c}, \quad (1),$$

где w_{ij} – значение элемента фильтра свёрточного слоя; X_{xyc} – область канала c , размерностью $x*y$, проходящего через фильтр $W_{cc'}$ свёрточного слоя.

Следует отметить наличие в формуле (1) двух гиперпараметров: 1) размерность фильтра $W_{cc'}$; 2) шаг свёртки (stride) – шаг, с которым берутся области X канала c при его прохождении через свёрточный слой. Ещё одним важным гиперпараметром в свёрточных слоях является пространственное дополнение (padding), позволяющее управлять размерностью каналов после их прохождения через свёрточный слой путём добавления им рамки с нулевыми значениями элементов. Значения данных гиперпараметров являются постоянными в пределах одного свёрточного слоя и подбираются эмпирическим путём.

Для того, чтобы фильтры в свёрточных слоях могли обучаться, в формуле (1) необходима нелинейность. В свёрточных слоях нелинейность достигается путём применения активационной функции ReLU, которая имеет следующий вид:

$$A(x) = \max(0, x). \quad (2)$$

График функции ReLU представлен на рисунке 1.

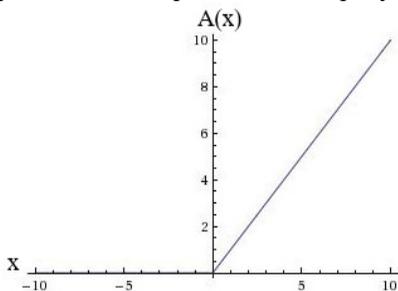


Рисунок 1 – График функции ReLU

Объединив формулы (1) и (2) получим окончательную формулу для нахождения значения элемента канала после его прохождения через фильтр свёрточного слоя:

$$O_{xye'}(x) = \max(0, x), \quad (3)$$

где $x = \sum_{i,j,c} W_{ijce'} X_{x-i,y-j,c}$

Общее количество обучаемых параметров в свёрточных слоях нейронной сети можно выразить следующей формулой:

$$W = \sum_{i=1}^M \frac{N_i * P_i * M_i * K_i'}{P_i}, \quad (4)$$

где N_i и K_i – размерность (с учётом пространственного дополнения) и количество каналов карты признаков, проходящей через свёрточный слой i , соответственно; M_i и K_i' – размерность и количество фильтров в свёрточном слое i соответственно; P_i – шаг свёртки в свёрточном слое i ; M – количество свёрточных слоёв.

Ещё одной важной операцией над картой признаков, проходящей через свёрточный слой, является пространственный пулинг (pooling). Пространственный пулинг — это операция, похожая на свёртку, но использующая вместо обучаемых фильтров статистическую функцию. Наиболее распространённый вид пулинга, используемый в свёрточных слоях — это max-pooling. Есть и другие варианты, например, mean-pooling, который применяет функцию усреднения по содержимому окна.

Математически функцию max-pooling можно выразить следующим образом:

$$O_{xye'}(X) = \max(X_{x,y,e'}; X_{x+x',y+y',e'}), \quad (5)$$

где $x' * y'$ - размерность окна и шаг пространственного пулинга.

Max-pooling позволяет уменьшить размерность каналов карты признаков после прохождения через свёрточный слой путём отбора самого значимого элемента в границах просматриваемого окна и отбрасывания остальных. Пространственный пулинг применяется не во всех свёрточных слоях, что делает его ещё одним гиперпараметром.

Далее рассмотрим принцип работы полносвязных слоёв. Каждый нейрон в таком слое соединён с каждым нейроном в последующем слое весовыми коэффициентами, которые являются обучаемыми параметрами. В качестве функции активации в полносвязных слоях свёрточной нейронной сети также используется ReLU. Согласно формуле (2) значение нейрона в полносвязном слое можно записать следующим образом:

$$O_{j'}(x) = \max(0, x), \quad (6)$$

где $x = \sum_{j=1}^N w_{ij} * X_j$; w_{ij} - коэффициент связи нейрона j в предыдущем слое с нейроном i в текущем слое; X_j - значение нейрона j в предыдущем слое; N - количество нейронов в предыдущем слое.

Общее количество обучаемых параметров в полносвязных слоях нейронной сети можно вычислить по следующей формуле:

$$W = \sum_{i=1}^M N_i * N_{i+1}, \quad (7)$$

где N_i - количество нейронов в слое i ; M - количество полносвязных слоев в нейронной сети.

В качестве базисной архитектуры для большинства современных свёрточных нейронных сетей, решающих задачу распознавания изображений, выступает модель VGG16 [2], победившая на всемирном конкурсе по распознаванию изображений ILSVRC-2014 [3].

Архитектура VGG16 представлена на рисунке 2.

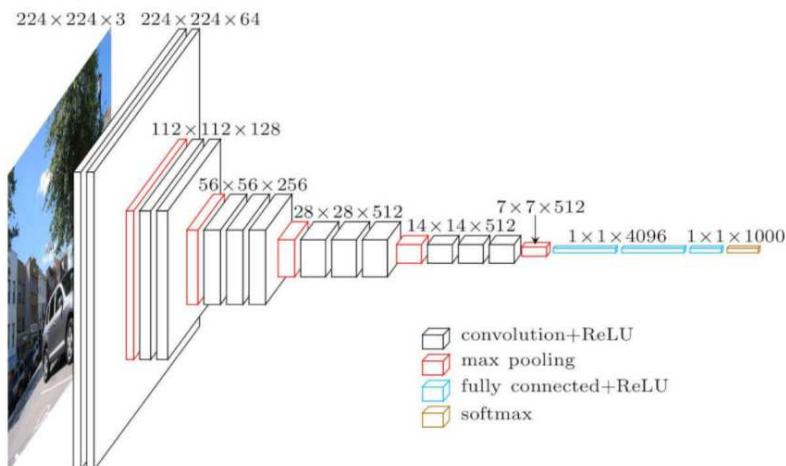


Рисунок 2 – Архитектура VGG16

Данная модель состоит из 16 слоёв. На вход ей поступают RGB изображения размерностью $224 * 224 * 3$. Первоначально изображения проходят через стек свёрточных слоев, в которых используются фильтры размерностью $3 * 3$. Stride и padding в VGG16 равняются 1. Все слои в сети снабжены ReLU (3). Пространственный пулинг осуществляется при помощи алгоритма max-pooling (5) окном $2 * 2$. После стека свёрточных слоев идут три полносвязных слоя: первые два имеют по 4096 нейронов, размерность третьего равна количеству классов распознаваемых изображений (на рисунке 2 третий полносвязный слой состоит из 1000 нейронов). В последнем слое на

выходе применяется функция soft-max, которая является обобщением логистической функции для многомерного случая [4]. Она преобразует значения нейронов выходного слоя в вероятностные показатели, характеризующие вероятности принадлежности входного изображения к тому или иному классу.

Приведём качественную оценку использования свёрточных нейронных сетей для решения задачи распознавания изображений. Показатель ошибки обучения VGG16 составляет приблизительно 6.8%. Количество обучаемых параметров, рассчитанное по формулам (5) и (7), равняется примерно 140 млн. Если убрать из сети на рисунке 2 все свёрточные слои и сразу подать на вход полносвязным слоям изображение размерностью $224 * 224 * 3$, то, согласно формуле (7), количество обучаемых параметров возрастёт до 637 млн. Таким образом, добавление 13 свёрточных слоёв ускорит процесс обучения нейронной сети в 4.55 раза. Кроме того, в задачах распознавания цветных изображений с большим разрешением количество слоёв является одним из параметров, благодаря которому можно уменьшить показатель ошибки обучения.

Подводя итог, можно сделать вывод, что применение нейронных сетей со свёрточной архитектурой является эффективным средством решения задач, связанных с распознаванием изображений, в том числе и задачи автоматизированного сбора данных об объектах по их фотографиям в информационных системах.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Hubel, D. H., & Wiesel, T. N. Brain mechanisms of vision. - Scientific American, 1979. – 241 с. – с. 150 – 162.
2. K. Simonyan, A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. – Published as a conference paper at ICLR 2015, 2015. - 14 с.
3. ImageNet Large Scale Visual Recognition Challenge 2014 (ILSVRC2014) [Электронный ресурс]. – Режим доступа: <http://www.image-net.org/challenges/LSVRC/2014/index>.
4. Ч. Аггарвал. Нейронные сети и глубокое обучение: учебный курс.: Пер. с англ. - СПб.: ООО "Диалектика", 2020. – 752 с. – с. 123 – 126.

УДК 004.51

В.Ю. КОЖИКОВ, Е.А. СЕНЬКИНАРязанский государственный радиотехнический университет
имени В.Ф. Уткина**МНОГОФУНКЦИОНАЛЬНЫЕ ПАНЕЛИ ОПЕРАТОРА**

Рассматриваются различные многофункциональные панели операторов и некоторые их элементы.

Радиоэлектронные устройства являются неотъемлемой частью современных приборов и оснащения различного применения. Их доля и значение неизменно возрастает. Также в данный момент увеличились требования к надежности службы электронных приборов, уменьшение их габаритов, стали более разнообразны варианты их конструктивного исполнения. К военной технике данные условия еще выше: широкий диапазон температур от -50°C до $+50^{\circ}\text{C}$, работа в жестких климатических условиях.

Установка многофункциональной панели оператора должна обеспечивать механическую прочность при низком атмосферном давлении и должна действовать на всем спектре температур, к которым она допущена. Панель оператора чаще всего представляет собой конструкцию, которая имеет плоскую переднюю часть с дисплеем и органами управления, защищенную от негативного воздействия окружающей среды и механического воздействия.

Внутреннее устройство панели аналогично устройству промышленного компьютера, с поправкой на особенности эксплуатации.

В общем случае можно сказать, что любая панель оператора располагает средством отображения информации: современная, чаще всего, в виде сенсорного текстового или графического экрана; в устаревших моделях применялись разнообразные электронные индикаторы: электронные табло и группы ламп или светодиодов, дальше — ЖК-дисплеи. Технические средства, которые обеспечивают выбор и ввод данных, а также навигацию по экранам: клавиатура, кнопки, сенсорный экран, встроенный или подключаемый джойстик, или манипулятор.

Можно выделить следующие важные характеристики панелей оператора:

1. Тип и размер экрана. Экран может быть разного разрешения, начиная от миниатюрного 128×128 и заканчивая 1024×768 . Экран

может быть, как цветным, так и монохромным, причём количество отображаемых цветов может варьироваться в диапазоне от 2 (монохромные панели) до более 16 млн (24 бит). Параметры отображения влияют на удобство восприятия информации и утомляемость обслуживающего персонала.

2. Организация управления. Тут вариантов немного: с помощью прозрачного сенсорного экрана (емкостного или резистивного) или с помощью функциональных кнопок и манипуляторов, расположенных на фронтальной стороне. Возможен и комбинированный вариант — например к экрану, обрабатывающему команду, программно добавляется кнопка аварийного останова процесса, которая действует на уровне электрической схемы.

3. Быстродействие процессора и объем встроенной Flash-памяти. Эти характеристики определяют максимальный объем прикладной программы визуализации и скорость обработки информации.

4. Количество интегрированных сетевых интерфейсов и поддерживаемых протоколов связи (зависит только от программного обеспечения, устанавливаемого производителем). Современные панели имеют встроенную поддержку сразу нескольких коммуникационных протоколов, например, Modbus RTU/TCP, Profibus DP и/или Industrial Ethernet, которые можно настроить на одном или нескольких интегрированных сетевых интерфейсах.

5. Степень защиты. Для фронтальной части это, как правило, IP65, для остальной части корпуса — IP20. Но бывают панели оператора со степенями защиты IP67 со всех сторон. Как правило, наибольшее значение имеет степень защиты именно фронтальной части, что связано с особенностью монтажа.

Близкими аналогами многофункциональной панели оператора можно назвать такие панели как: SIMATIC MP 177, MP 277 и MP 377 (Siemens), WebOP-2121 (Advantech). Отечественные близкие аналоги производит АО Плазма – ПО.

Многофункциональные панели SIMATIC MP x77 (рисунок 1.) являются платформой для построения систем человеко-машинного интерфейса и сочетают в себе черты панелей оператора, промышленных компьютеров и программируемых контроллеров. Небольшая монтажная глубина, отсутствие жестких дисков и вентиляторов, степень защиты фронтальной панели IP65 и длительный срок службы ламп подсветки позволяют применять панели MP x77 в жестких промышленных условиях. Операционная система Windows CE и встроенное программное обеспечение панелей MP x77 обеспечивают поддержку стандартных интерфейсов Windows.



Рисунок 1 – Многофункциональные панели SIMATIC MP x77

Фронтальная часть корпуса всех панелей имеет степень защиты IP65, остальная часть корпуса имеет степень защиты IP20. Панель оператора MP 277-10 Touch INOX ориентирована на применение в пищевой и фармацевтической промышленности, а также на предприятиях по производству напитков. Ее фронтальная панель выполнена из нержавеющей стали и имеет степень защиты IP66. По своим установочным размерам, набору встроенных интерфейсов и объему поддерживаемых функций эта панель аналогична панели MP 277-10 Touch.

Многофункциональные панели WebOP-2121 фирмы Advantech (рисунок 2.) спроектированы с RISC-процессором на базе ARM9 с flash-памятью 70 ~ 200МГц и 4 ~ 16Мб для прикладного программного обеспечения. Серия WebOP-2000V также поддерживает различные размеры ЖК-дисплея от 3,5 "до 12,1" для различных приложений, связанных с использованием контроллеров температуры, движения, инверторов и датчиков.



Рисунок 2 – Многофункциональная панель WebOP-2121

Из отечественных аналогов – компании «Плазма» выпускает панель оператора (рисунок 3) на основе монохромной газоразрядной индикаторной панели оранжевого цвета свечения с информационной емкостью 240×120 элементов отображения, с шагом элементов отображения (по горизонтали и вертикали) $0,45\text{мм}$ предназначена для самоходного артиллерийского орудия (CAO). Взаимодействие программного обеспечения с внешними устройствами осуществляется в соответствии с установленным программным обеспечением по сигналам внешних интерфейсов CAN (2.0 В ISO-11898) (2 независимых канала), RS-232 (ГОСТ Р 50668), RS-485 (EIA RS-485), подаваемым на входные разъемы.



Рисунок 3 – Панель оператора

В современной радиоэлектронике идет тенденция на снижение массы изделия, получение высокой степени монтажа и обеспечение герметичности. Эти факторы привели к понижению жесткости элементов конструкции. По статистике считают, что 40-45% отказов происходит из-за ошибок, допущенных на этапе проектирования, 20%- из-за несовершенства технологического процесса производства, 30%- в следствии неправильного выбора режима эксплуатации, а оставшиеся проценты приходятся на естественный износ составных частей и старение материала.

При производстве МПО большую роль играют такие конструктивные элементы, как экран и система охлаждения.

В многофункциональных панелях оператора для производства лицевой стороны используется защитное стекло. При выборе стекла руководствуются следующими требованиями:

- Высокая механическая прочность
- Твердость
- Низкая плотность
- Абразивоустойчивость

Для охлаждения РЭА разработано несколько способов: жидкостный, воздушный, испарительный, кондуктивный, термоэлектрический.

Одним из самых простых является метод охлаждения с помощью радиатора. Этот метод основан на явлениях теплопроводности материалов и естественной конвекции. Обдув радиатора воздушной струей от вентиляторов решает эту проблему и значительно усиливает теплоотвод.

Успешнее с данной задачей справляются жидкостные охладители, которые имеют более высокую теплоемкость. Принцип работы системы охлаждения можно описать следующим образом: миниатюрный резервуар закрепляется на поверхности охлаждаемого чипа; через шланг, с помощью микропомпы охлаждающая жидкость перекачивается в герметичный наружный радиатор, который в свою очередь, может обдуваться струей воздуха от специального вентилятора.

Другой современной системой охлаждения является термоэлектродохладитель, принцип работы которого основан на эффекте Пельтье. Эффект основан на подаче постоянного тока в цепь, состоящую из двух проводников, и в местах контакта тепло будет поглощаться или выделяться в зависимости от направления тока. Бывают ситуации, что для более эффективного теплоотвода используется не один модуль, а несколько, из которых выстраивается

каскад. В этом случае на горячую поверхность первого модуля устанавливается второй чуть большего размера. Иногда собирается целая пирамида модулей. С ее помощью можно достичь разницы температур в 70-75°C при температуре окружающей среды 25°C.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Михеев М. А., Михеева И. М. Основы теплопередачи. М., Энергия, 1973. – 320 с.
2. Уонг Х. Основные формулы и данные по теплообмену для инженеров: Пер. с англ. / Справочник. – М.: Атомиздат, 1979. – 216 с.
3. Пошехонов П.В., Соколовский Э.И. Тепловой расчет электронных приборов. – М.: Высшая школа, 1977. – 160 с.
4. Басов А. М., Быков В. Г. Электротехнология – М.: Агропромиздат 1985 – 1985 -256с.
5. Краснощекова Г.Ф., Зеленский А.В. Анализ конструкций приборов РЭС: Метод. указ. ИПО СГАУ 2007г. 20 с.
6. Конструирование радиоэлектронных средств. Разработка функциональных узлов на печатных платах: Методическое указание к лабораторной работе /Рязан.гос.радиотех.акад.Сост. Румянцев В.П.,Мухин В.П., Нелидкин А.М.Рязань,1998.- 20с.
7. Осипова В.А. Экспериментальное исследование процессов теплообмена. – М., «Энергия», 1969.
8. ГОСТ 2.105-95 (2002) «Общие требования к текстовым документам».
9. Бородин С.М. Обеспечение тепловых режимов в конструкциях радиоэлектронных Ульяновск 2008.
10. С. Чи Тепловые трубы теория и практика. - М.: Машиностроение, 1998.
11. Тимошенко С.П., Войновский-Кригер С. Пластины и оболочки. – М.:Наука, 1966. – 636 с.

УДК 004.021

Э.Г. КОЧЕРГИН, А.А. ТРУБИЦЫН
Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

ДВИЖЕНИЕ ПАРАКСИАЛЬНЫХ ПУЧКОВ ЭЛЕКТРОНОВ В АКСИАЛЬНО-СИММЕТРИЧНОМ ПОЛЕ

Рассматриваются краткие теоретические сведения относительно уравнения движения параксиальных электронов в аксиально-симметричном электростатическом поле и его практического применения при траекторном анализе в иммерсионной линзе.

Электронная оптика первого порядка.

Электронная оптика – наука о формировании, фокусировке и транспортировке потоков заряженных частиц, в электрических и магнитных полях. Важнейшим понятием, затронутой науки, является электронно-оптические системы ЭОС, позволяющие фокусировать пучки заряженных частиц.

Различают аналитические и численные методы проектирования ЭОС. В основе аналитических методов лежат подходы классической электронной оптики или параксиальной электронной оптики (электронной оптики первого порядка). В этот раздел оптики входит понятие параксиальной области, т.е. области бесконечно приближенной к оптической оси.

Для моделирования ЭОС необходимо рассмотреть аксиально-симметричные системы. Аксиально-симметричные поля – это такие поля, функция которых зависит только от расстояния до оси и имеет следующий вид $f = f(r)$. Эти поля можно назвать цилиндрическими. Для их изучения необходимо использовать цилиндрическую систему координат.

Моделирование электронно-оптической системы включает в себя несколько этапов. Одним из таких этапов является расчет траекторий заряженных частиц в аксиально-симметричном поле.

Уравнение Буша.

Рассмотрим линейное дифференциальное уравнение второго порядка относительно $r = r(z)$, т.е. уравнение Буша.

$$\frac{d^2 r}{dz^2} + \frac{1}{2} \cdot \frac{\Phi_0'(z)}{\Phi_0(z)} \cdot \frac{dr}{dz} + \frac{1}{4} \cdot \frac{\Phi_0''(z)}{\Phi_0(z)} \cdot r = 0, \quad (1)$$

где $\Phi_0(z) = U(z, 0)$ – осевое распределение потенциала, т.е. распределение потенциала вдоль оси симметрии поля.

Это уравнение впервые было получено Г. Бушем в 1926 году.

Данное уравнение представляет собой уравнение движения парааксиальных электронов в аксиально-симметричном электростатическом поле т.е. поле обладающем осевой симметрией. Оно связывает координаты электрона z и r .

Уравнение Буша часто в различной литературе называют основным уравнением электронной оптики электростатических полей.

Проанализировав уравнение (1), мы можем заметить следующие моменты:

1. В уравнение Буша входит только осевое распределение потенциала $\Phi_0(z)$. Значит, чтобы определить траекторию движения парааксиальных частиц в аксиально-симметричном электростатическом поле, нам не обязательно знать распределение потенциала во всем пространстве. Нам необходимо знать только осевое распределение потенциала $\Phi_0(z)$, найденное тем или иным способом [1].

2. В уравнении Буша не присутствует ни заряд, ни масса движущейся парааксиальной частицы в аксиально-симметричном электростатическом поле, поэтому данный закон можно использовать для расчета траекторий движения не только электронов, но и любых других заряженных частиц, например, более тяжелых отрицательных ионов [1].

3. Уравнение (1) является линейно однородным дифференциальным уравнением второго порядка относительно $\Phi_0(z)$. Потенциал $\Phi_0(z)$ и его производные $\Phi_0'(z)$ и $\Phi_0''(z)$, присутствуют в первой степени и правая часть уравнения равна 0. Отсюда следует, что при изменении потенциалов всех электродов, создающих поле, изменить в равное число раз, то траектория движения парааксиальных частиц не изменится. Следовательно, мы можем проводить исследования электронных линз на моделях с пропорционально измененными напряжениями на всех электродах или их подключением к цепям с переменным током [1].

4. Данное уравнение также является линейно однородным относительно $r(z)$. Отсюда тоже следует, что, если использовать модель электродов электронно-оптической системы, увеличенную в определенное количество раз, то и траектория движения будет увеличена, но останется геометрически подобной траектории в первоначальной системе [1].

Иммерсионная линза.

Иммерсионная линза – линза, к которой с обеих сторон примыкают области с различными постоянными потенциалами. Такая линза имеет две области преломления. В основном, линза работает как собирающая.

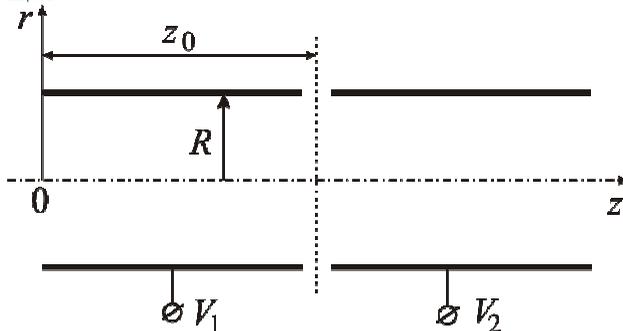


Рисунок 1 – Иммерсионная линза

Осевое распределение потенциала $\Phi_0(z)$ в иммерсионной линзе, состоящей из двух цилиндров с потенциалами V_1 и V_2 радиусом R (Рисунок 1) может быть описано приближенным выражением (2).

$$\Phi_0(z) = \frac{V_2 + V_1}{2} + \frac{V_2 - V_1}{2} \operatorname{th} \left(\frac{1,315(z - z_0)}{R} \right), \quad (2)$$

где z_0 – координата зазора между цилиндрами.

Траекторный анализ.

Проведем исследование точности вычисления траектории с помощью уравнения Буша (1) в иммерсионной линзе. Для этого воспользуемся средой MathCAD и программой моделирования аксиально-симметричных электронно-оптических систем «ФОКУС».

Представим уравнение Буша в следующем виде:

$$\frac{d^2 r}{dz^2} = -\frac{1}{2} \cdot \frac{\Phi_0'(z)}{\Phi_0(z)} \cdot \frac{dr}{dz} - \frac{1}{4} \cdot \frac{\Phi_0''(z)}{\Phi_0(z)} \cdot r. \quad (3)$$

Перед началом вычислений, зададим все необходимые первоначальные параметры (Рисунок 2), такие как:

1. Радиус цилиндров – R ;
2. Координату центра зазора – z_0 ;
3. Потенциалы левого и правого цилиндров – V_1 и V_2 .

1. Задаем потенциалы электродов и геометрические параметры линзы

$$V1 := 1 \quad z0 := 4$$

$$V2 := 7 \quad R := 1$$

Рисунок 2 – Задание первоначальных данных в среде MathCAD

Следующим шагом задаем функцию осевого распределения потенциала $\Phi_0(z)$ и строим график этой функции.

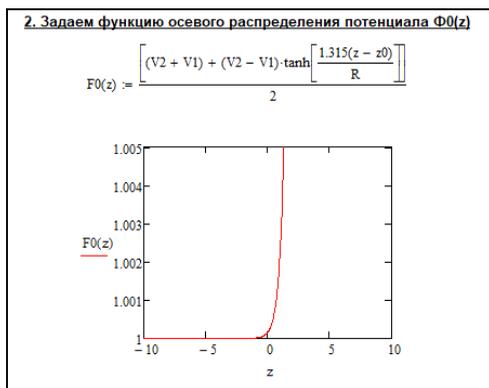


Рисунок 3 – Функция осевого распределения в среде MathCAD

Далее для решения линейного дифференциального уравнения Буша (3), нам необходимо задать функции, определяющие его правую часть. Также строим графики зависимости этих функций.

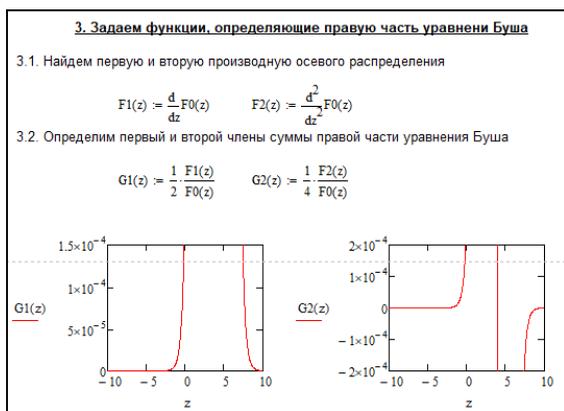


Рисунок 4 – Функции, определяющие правую часть дифференциального уравнения в среде MathCAD

Заключительным этапом построения траекторий в среде MathCAD, является решение дифференциальных уравнений с помощью блока Given...Odesolve.

В случае параксиальных пучков траектории являются пологими, т.е. $dr/dz \ll 1$. Угол наклона движения электронов на всей протяженности траектории к оси симметрии z очень мал. Поэтому, проведем вычисление траекторий с нулевым начальным углом ($r'=0$). Траектория (Рисунок 6) имеет начальные координаты ($z; r$) равные (0; 0,1).

```

4. Решаем Дифференциальное уравнение с начальными условиями

Given

r''(z) = -G1(z)·r'(z) - G2(z)·r(z)

r(0) = 0.1
r'(0) = 0.0

r := Odesolve(z,9)

z := 0, 0.01.. 8

```

Рисунок 5 – Решение дифференциального уравнения в среде MathCAD

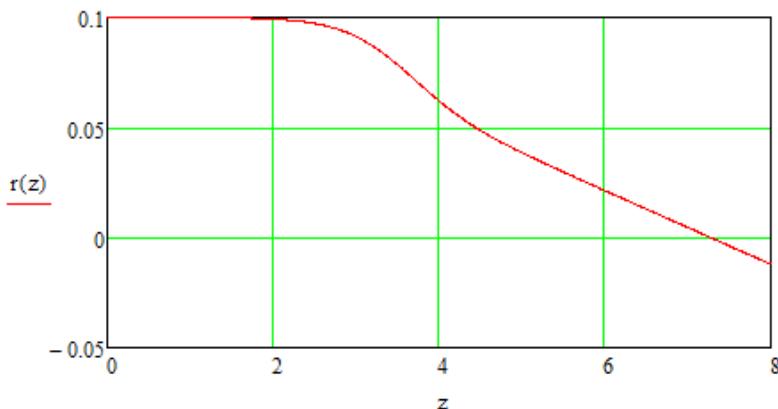


Рисунок 6 – Траектория движения электронов в среде MathCAD

Расчитаем траекторию в программе «ФОКУС», для сравнения с расчетами, произведенными в среде MathCAD. Программа

«ФОКУС» необходима для моделирования аксиально-симметричных электронно-оптических систем. Программа состоит из нескольких модулей (графический редактор, модуль вычисления электрического поля, модули траекторного анализа).

Для начала расчета сформируем в модуле Design (модуль графического редактора) конструкцию линзы.

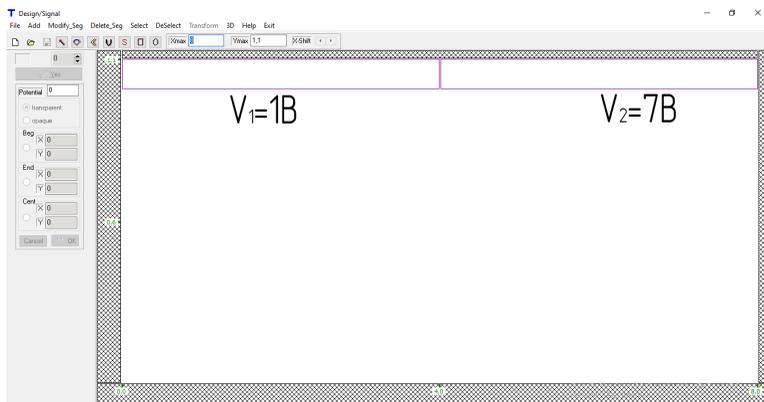


Рисунок 7 – Сечение конструкции иммерсионной линзы в программе «ФОКУС»

В модуле Path_S (модуль моделирования траекторий заряженных частиц в электростатическом поле) рассчитаем траекторию с начальными условиями: $E=1$, $AngBeg=0$, $AngEnd=0$, положение источника для первой траектории $X=0$, $Y=0.1$.

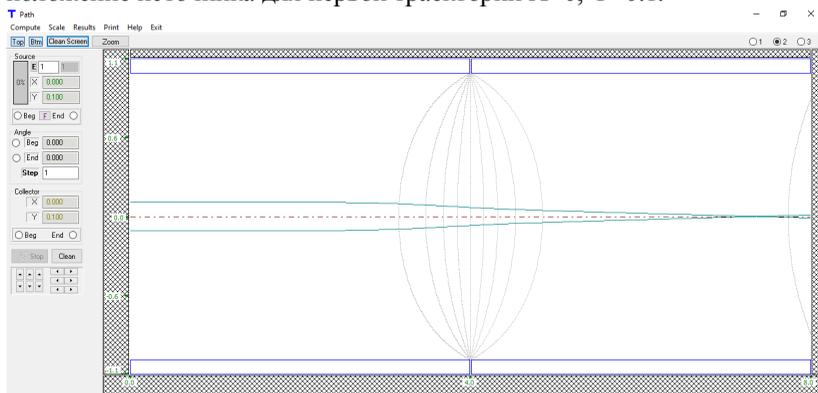


Рисунок 8 – Результат траекторного анализа в программе «ФОКУС»

Таким образом, мы построили траекторию движения частиц в иммерсионной линзе с заданными параметрами и начальными условиями с помощью среды MathCAD и программы «ФОКУС».

Сравнив траекторию, построенную в среде MathCAD с траекторией, построенной в программе «ФОКУС», можно сказать, что расчеты произведенные в MathCAD в достаточной степени точности отображают реальный характер траектории движения частиц в аксиально-симметричном электростатическом поле. Данную программу, можно использовать для произведения подобных расчетов.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Шерстнев Л. Г., Электронная оптика и электроннолучевые приборы. Учебник для студентов высш. техн. учебн. Заведений, М., Энергия, 1971.
2. FocusPro : [сайт]. URL: <https://focuspro-soft.com/ru/>

УДК 004.9

А.Д. КОШЕЛЕВ

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ В ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКЕ

В статье рассматривается суть генетического алгоритма, основные понятия ГА, приводится его классификации (последовательный и параллельный), а также примеры области применения.

Генетические алгоритмы – класс вычислительных моделей, которые используют механизмы естественной эволюции для решения задач в разнообразных областях. Теория ГА была описана Д. Холландом в книге "*Адаптация в естественных и искусственных системах*", в 1975 году.

Такие алгоритмы параллельны, т.к. их поиск наилучшего решения осуществляется по генетическим структурам (строительным блокам), представляющих ряд возможных решений. ГА состоит из набора отдельных элементов (популяции) и набора биологически вдохновленных операторов, определенных для самой популяции.

В вычислительных терминах генетический алгоритм отображает задачу на набор (двоичных) строк, каждая из которых представляет потенциальное решение. ГА использует самые оптимальные строки для поиска улучшенных решений. Данные метод обычно проходит через цикл из 4-х стадий (см. рисунок 1).

В каждом цикле вырабатывается *новое поколение* возможных решений данной задачи. На первом этапе создается *начальная совокупность* потенциальных решений, как отправная точка процесса поиска. Каждый элемент популяции *закодирован* в цепочку (*хромосому*), которой должны управлять *генетические операторы* (варианты процессов рекомбинации и генетических изменений, адаптированные к требованиям конкретных задач, инверсия, доминирование, рекомбинация генетических краев и т. д.)

На следующем этапе оценивается работоспособность (или *приспособленность*) каждого индивида популяции с учетом ограничений, налагаемых данной проблемой. Основываясь на приспособленности каждого индивидуума, механизм отбора выбирает "партнеров" для процесса *генетических манипуляций*. Политика отбора

в конечном счете отвечает за обеспечение выживания наиболее приспособленных особей.

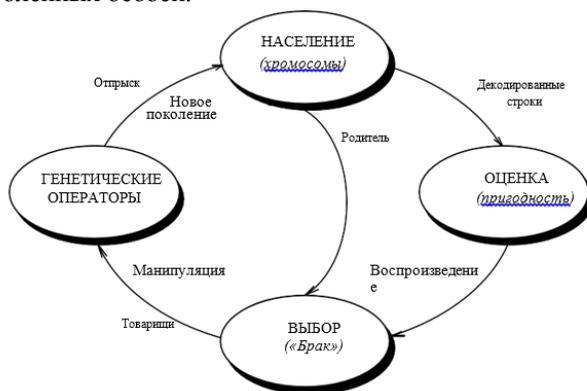


Рисунок 1 – Цикл ГА

Процесс манипуляции использует генетических операторов для создания новой популяции особей (*потомства*) путем манипулирования «генетической информацией», называемой *генами*, которой обладают члены (*родители*) текущей популяции. Она включает в себя две операции: скрещивание и мутацию. *Кроссинговер* отвечает за рекомбинацию генетического материала популяции. Процесс отбора, связанный с рекомбинацией, гарантирует, что генетические структуры, сохраняются для будущих поколений. Тогда строительные блоки представляют собой наиболее подходящие генетические структуры в популяции. Тем не менее, сам по себе процесс рекомбинации не может избежать потери перспективных строительных блоков в присутствии других генетических структур, что может привести к локальным минимумам. Кроме того, он не способен исследовать участки пространства поиска, не представленные в генетических структурах популяции. Затем в действие вступает оператор мутации. Он вводит новые генетические структуры в популяцию путем случайной модификации некоторых ее строительных блоков. Это помогает алгоритму поиска избежать ловушек локальных минимумов. Модификация, вносимая оператором мутации, не связана с какой-либо предшествующей генетической структурой популяции, она позволяет создавать различные структуры, представляющие другие участки пространства поиска.

Оператор скрещивания берет две хромосомы и обменивает часть их генетической информации, чтобы произвести новые хромосомы. После *того*, как точка пересечения была выбрана случайным образом,

части *родительских* строк P1 и P2 меняются местами, чтобы произвести новые *дочерние* строки O1 и O2. Например, на рисунке 2 показан оператор Crossover, применяемый к пятому и шестому элементам строки.

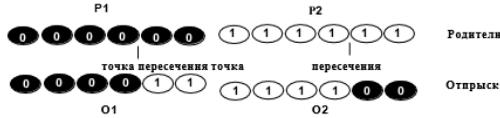


Рисунок 2 - Crossover

Рисунок 2 – Кроссовер

Мутация реализуется путем случайного изменения а случайного бита в а строке. На рисунке 3 представлен оператор мутации, применяемый к четвертому элементу строки.

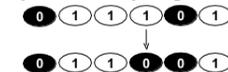


Рисунок 3 – Мутация

Потомство, полученное в процессе генетических манипуляций, образует следующую популяцию, подлежащую оценке. ГА могут либо заменить всю популяцию (подход поколений), либо только их менее приспособленные члены (стационарный подход). Цикл *создания-оценки - отбора-манипуляции* повторяется до тех пор, пока не будет найдено удовлетворительное решение проблемы.

Для иллюстрации реализации последовательного ГА воспользуемся примером простой оптимизации функций, приведенным в работе Гольдберга, и рассмотрим его программирование на языке С.

Первый шаг в оптимизации функции $f(x)=x^2$, на интервале [0-31], заключается в кодировании набора параметров x , например, в виде пятизначной двоичной строки {00000–11111}. Затем генерируем начальную популяцию из 4 потенциальных решений, показанных в таблице 1, с помощью генератора случайных чисел.

Таблица 1.

Начальная популяция	x	Полезность	%
01101	13	169	14.4
11000	24	576	49.2
01000	8	64	5.5
10011	19	361	30.9
Итого		1170	100

Объявляем группу популяций в виде массива с четырьмя элементами, как показано на рисунке 4, а затем инициализируем структуру с помощью генератора случайных чисел, как показано на рисунке 5.

```
#define POPULATION_SIZE 4 /* Size of the population */
#define CHROM_LENGTH 5 /* String size */
#define PCROSS 0.6 /* Crossover probability */
#define PMUT 0.001 /* Mutation probability */

struct population
{
    int value;
    unsigned char string[CHROM_LENGTH];
    int fitness;
};
struct population pool[POPULATION_SIZE];
```

Рисунок 4 – Объявления глобальных констант и переменных в языке C

```
initialise_population()
{
    randomise(); /* random generator set-up */
    for (i=0; i < POPULATION_SIZE; i++)
        encode(i, random(pow(2.0, CHROM_LENGTH)));
}
```

Рисунок 5 – Процедура инициализации

После инициализации ГА, следующим этапом является воспроизведение. Репродукция оценивает и выбирает пары строк для *спаривания* – например, с помощью метода "колеса рулетки" – в соответствии с их относительной силой (см. таблицу 1 и соответствующий код C на рисунке 6). Выбирается одна копия строки 01101, две копии строки 11000 и одна копия строки 10011.

```
select(sum_fitness)
{
    parsum = 0;
    rnd = rand() % sum_fitness; /* spin the roulette */
    for (i=0; i < POPULATION_SIZE; parsum <= rnd; i++)
        parsum += pool[i].fitness; /* look for the slot */
    return (-i); /* returns a selected string */
}
```

Рисунок 6 – Функция выбора

Далее мы применим оператор Crossover. Crossover работает в два этапа (см. рисунок 7). Во-первых, он определяет, будет ли Crossover происходить на паре строк с помощью функции flip; бросание предвзятой монеты. Если heads (true) с вероятностью *pcross*, строки меняются местами; *crossover_point* определяется генератором случайных чисел. Если tails (false) строки просто копируются. В примере Crossover происходит на пятой позиции для первой пары и на третьей позиции для другой.

```

crossover (parent1, parent2, child1, child2)
{
  if (flip(PCROSS))
  {
    crossover_point = random(CHROM_LENGTH);
    for (i=0; i <= CHROM_LENGTH; i++)
    {
      if (i <= site)
      {
        new_pool[child1].string[i] = pool[parent1].string[i];
        new_pool[child2].string[i] = pool[parent2].string[i];
      }
      else
      {
        new_pool[child1].string[i] = pool[parent2].string[i];
        new_pool[child2].string[i] = pool[parent1].string[i];
      }
    }
  }
}

```

Рисунок 7 – Процедура Crossover

После Crossover оператор мутации применяется к новой популяции, которая может иметь случайный бит в данной строке измененной. Функция *мутации* на рисунке 8 использует смещенный бросок монеты (*flip*) с вероятностью *pmut*, чтобы определить, нужно ли немного изменить или нет.

```

mutation ()
{
  for (i=0; i < POPULATION_SIZE; i++)
  {
    for (j=0; j < CHROM_LENGTH; j++)
    {
      if (flip(PMUT))
        pool[i].string[j] = ~new_pool[i].string[j] & 0x01;
      else
        pool[i].string[j] = new_pool[i].string[j];
    }
  }
}

```

Рисунок 8 – Реализация оператора мутации *S*

Теперь была создана новая популяция (см. таблицу 2), и применяется тест завершения. Критерии завершения могут включать: время моделирования, превышенное число поколений или удовлетворенный критерий сходимости. В этом примере мы могли бы установить число поколений до 50, а сходимости как среднее улучшение пригодности менее 5%, между поколениями. Для начальной популяции среднее значение составляет 293 (т. е. $(169+576+64+361) \div 4$), в то время как для нового населения она увеличилась до 439 (то есть 66%).

Таблица 2.

Начальная популяция	x	Полезность	%
01100	12	144	8.2
11001	25	625	35.6
11011	27	729	41.5
10000	16	256	14.7
Итого			100

Помимо внутреннего параллелизма, вычислительные модели ГА также могут использовать несколько уровней параллелизма, благодаря естественной независимости операций генетической манипуляции.

Параллельный ГА обычно формируется *параллельными компонентами*, каждый из которых отвечает за манипулирование суб-популяциями. Существует два класса ПГА, использующих *централизованные* и *распределенные* механизмы выбора. С централизованными механизмами отбора единый оператор отбора работает с глобальной популяцией (суб-популяций). Таким образом, ПГА имеет синхронный этап выбора.

С распределенными механизмами каждый параллельный компонент имеет свою собственную копию оператора выбора. Кроме того, каждый компонент связывает свои лучшие строки с подмножеством других компонентов. Это поддерживается оператором *миграции* и *скоростью миграции*, определяющей интервал связи. Эти ПГА имеют асинхронный этап выбора. Алгоритм ASPARAGOS является отличным примером этого класса.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. L. Davis "Handbook of Genetic Algorithms", Van Nostrand Reinhold, New York – 1991.
2. K.A. De Jong "An Analysis of the Behavior of a class of Genetic Adaptive Systems", PhD thesis, University of Michigan – 1975.
3. D. E. Goldberg. "Genetic Algorithms in Search, Optimization & Machine Learning", Addison- Wesley Publishing Company] – 1989.
4. M. Gorges-Schleuter "ASPARAGOS An Asynchronous Parallel Genetic Optimisation Strategy", in H. Schaffer, editor, 3rd International Conference on Genetic Algorithms, pp. 422-427, Morgan Kaufmann – 1989.
5. J. J. Grefenstette "GENESIS: A System for Using Genetic Search Procedures" in Proceedings of the 1984 Conference on Intelligent Systems and Machines, pp. 161-165 – 1984.
6. F. Hoffmeister & T. Bäck "Genetic Algorithms and Evolution Strategies: similarities and differences" in Technical Report "Grüne Reihe" No. 365, Department of Computer Science, University of Dortmund, Germany – 1990.

УДК 622.013

М.К. КРЫГИНА, А.А. ЧУСОВА

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

ВИЗУАЛИЗАЦИЯ СЕЙСМОАКУСТИЧЕСКОЙ ИНФОРМАЦИИ

В данной статье будет рассмотрен комплекс технических средств сбора и регистрации информации морского геофизического комплекса на объекте эксплуатации.

Высокий темп роста объема добычи полезных ископаемых привел к увеличению глубины их разработки и к усложнению геологоразведки.

Сейсмоакустика является комплексом разночастотных методов, которые основываются на использовании упругих волн (акустических, сейсмических или ультразвуковых). Для более детального изучения физико-механических свойств грунтовых массивов применяются сейсмоакустические исследования.

Сейчас большое распространение находит использование разнообразных методов сейсмоакустических исследований для добычи полезных ископаемых. Также вышеупомянутые исследования используются для выявления и картирования потенциально опасных геологических объектов, например, накопления взрывоопасных газов, затопления, овражной и струйчатой эрозии и т.д.

Разработка полезных ископаемых таит в себе опасность аварий различного масштаба, многие из которых возможно было бы предотвратить, если было бы лучше исследовано расположение препятствий на объекте разработки.

Для предсказания различного рода геологических опасностей используются различные сейсмические исследования, например, сейсмологические наблюдения, акустические исследований на образцах грунта и т.д. Сами сейсмические исследования сейчас классифицируют на высокоразрешающую сейсморазведку, сейсмоакустические и акустические исследования.

Высокоразрешающая сейсморазведка основывается на регистрации сигнала в широкой полосе частот. Фиксируется разрез (часть его, находящаяся ближе к поверхности) на глубине от нескольких сотен метров до километра. При этом разрешающая способность должна быть не хуже пяти метров. Для повышения качества исследований используются более высокочастотные

источники, а также сравнительно небольшие многоканальные косы, длина которых достигает всего лишь до 1 км, а шаг наблюдений – 5 м.

Акустические исследования предполагают работу с акустическими профилографами и гидролокаторами бокового обзора. Источник и приемник упругих волн находятся в одной и той же скважине. Изучению подвергаются волны, которые распространяются вдоль скважины.

Достоинства сейсмоакустических исследований – простота измерений и установки геофонов, возможность надежно и в краткие промежутки времени получить непрерывную интегральную информацию о процессах, происходящих при выполнении работ по подготовке и добыче. Метод используют для слежения за динамикой образования трещин, изменения почвы, прогнозов динамических проявлений горного давления и т.д.

Цель сейсмоакустических исследований – изучение геологического разреза, глубина которого достигает до 100 м (разрешающая способность 0.5 – 5 м). Однако, гидроакустические приборы не могут обеспечить данной глубины исследований.

Для обеспечения наибольшей разрешающей способности и детальности при сейсмоакустических исследованиях на акваториях используют импульсные источники упругих волн на высоких частотах (высокочастотные) и небольшие буксируемые приемные системы, количество каналов которых может быть равно одному, а может и достигать 16 – 48 каналов. Источник и приемник располагаются на небольших глубинах (до одного метра). Однако, это порождает проблему в назначении такой малой глубины и не может обеспечить нормальную работу при неблагоприятных погодных условиях (даже при сильном течении работа уже становится трудновыполнимой, а в шторм такие исследования невозможны).

Современная геофизическая техника предполагает использование ЭВМ для цифровой регистрации и обработки информации, измерения на больших глубинах, а также в экстремальных климатических условиях, что позволяет по-новому подойти к самому процессу получения и обработки геофизической информации.

Морские сейсморазведочные исследования могут выполняться на шельфе с разнообразной глубиной вплоть до мелководья, в бухтах, заливах и т.д. Для работы в различных условиях используются мобильные сейсморазведочные системы, которые можно демонтировать с одного судна на другое.

Выделяют 2 класса методов морской сейсморазведки. Их сходство заключается в способах получения данных, а различие – в задачах, которые поставлены перед исследователями. Нефтяная сейсморазведка, которую можно отнести к первому классу, включает сейсмоисследования, целью которых является поиски морских залежей нефти и газа и их дальнейшая разработка. Второму классу (инженерная сейсморазведка) включает исследование верхних неконсолидированных осадков на глубинах от 100 до 500 метров (иногда может достигать километра). Одна из целей таких исследований - планирование строительства крупных морских сооружений.

Существуют амплитудные аномалии на сейсмических разрезах, которые связаны с газонасыщением пород, и необходимо выбрать безопасную площадку для установки буровой платформы

Выбросы газа при попадании бурового инструмента даже в очень маленькие скопления в отложениях, могут приводить к различным аварийным ситуациям: пусть даже не произойдет взрывной выброс с непосредственной аварией буровой платформы, но частичная утечка газа может привести к падению давления внутри пластов, и в следствие, к оседанию грунта. Если вдруг в этом месте будет находится одна из опор буровой платформы, то авария будет неизбежна.

Сейсмические исследования в морской зоне выполняются с использованием различных комплексов, регистрирующих сейсмоакустическую информацию. В качестве примера такого комплекса рассмотрим «DigiSTREAMER 2D», разработанный компанией «ION» Corp. (США).

«DigiSTREAMER 2D» - это интегрированная система для получения сейсмических данных о морском ландшафте. Данная система обеспечивает высокое качество морской сейсморазведки и отвечает требованиям по защите окружающей среды. Надежная конструкция системы дает возможность работать при любых, даже неблагоприятных погодных и природных условиях.

Регистрирующий комплекс имеет ряд особенностей:

1. Непрерывная запись данных (сейсмических), обеспечивающая получения всей информации, не ограничиваясь установленным периодом регистрации;

2. Аналоговый фильтр высоких частот (2 Гц), обеспечивающий максимальную ширину регистрируемого диапазона частот;

3. Системы массивов дисков для хранения данных (RAID), обеспечивающие надежную запись данных и эффективную защиту от их утери;

4. Удобный интерфейс пользователя, также повышающее эффективность работы.

Центральная электроника DigiSTREAMER – это платформа для получения морских сейсмических данных с использованием морских кос ION с гелевым наполнителем.

Центральная электроника и полевые модули (Streamer Electronic Modules) поддерживают до 20 кос и 19 200 каналов при дискретизации 1 мс.

Современный интерфейс оператора и разработанный механизм контроля качества позволяет быстро и эффективно контролировать все заборное оборудование. Система поддерживает так называемую концепцию «одним взглядом», позволяющую моментально находить ошибки, как заборного оборудования, так и центральной электроники.

В обобщении вышесказанного выделим основные преимущества комплекса «DigiSTREAMER 2D»:

1. Возможность работы со значительным количеством кос
 2. Удобный пользовательский интерфейс, облегчающий работу с продуктом
 3. Комплекс может быть обслужен всего лишь одним оператором, благодаря своему удобству
 4. Контроль качества исследований на различных уровнях:
 - 4.1. Отслеживание данных в реальном времени
 - 4.2. Слежение за состоянием электроники
 - 4.3. Тестирование рабочих модулей
 5. Системы спутниковой навигации и синхронизации времени
- Активная секция косы DigiSTREAMER.

В большинстве случаев исследования морского рельефа выполняются с плавучими косами, установленными на больших судах, которые должны обладать следующими характеристиками: иметь широкую корму, на которую сматанные буксируемые косы, источники колебаний, компрессоры и иное вспомогательное оборудование. С кормы производится погружение основного заборного оборудования (косы) и сейсмических источников в воду.

Плавающая цифровая сейсмическая коса – это секционированный шланг из пластика, который наполнен гелем или керосином, и включает в себя приборные и вспомогательные секции, соединенные герметичными муфтами. Длина косы – 9-12 км.

Приборные секции рассчитаны на 12 каналов и достигают длины 80-150 м. В них находятся геофоны и модуль сбора данных.

Активная секция косы предназначена для сбора морских сейсмических данных. Включает две катушки FSK, установленные в зафиксированном положении в передней и задней частях косы.

Характеристики косы можно выделить следующие:

1. Оснащена набором датчиков со схемой гидрофонов. Они обеспечивают понижение шума
2. Материал наполнения, имеющий гелеобразную структуру, позволяет снизить собственный шум
3. Для устранения помех от ускорений датчиков используются гидрофоны Teledyne T-2BX
4. Элемент напряжения типа Vectran® обеспечивает высокую прочность на растяжение и прогнозируемое удлинение при длинной буксируемой компоновке
5. Элементы, выполненные из титана, позволяют снизить вес и добиться износостойкости оборудования.
6. Максимальное использование готовой коммерческой технологии для снижения затрат и повышения надежности

Однако, существует и аналоги комплексов визуализации сейсмоакустической информации. Остановимся подробнее общих требованиях к продукту, предназначенному для решения поставленной задачи.

1. Функциональное назначение:

1.1. прием первичной сейсмоакустической информации, навигационных параметров, прием и фиксация момента срабатывания источника упругих колебаний;

1.2. тестирование секций сейсмокосы;

1.3. управление секциями сейсмокосы.

2. Область применения комплекса – комплекс технических средств сбора и регистрации информации геофизического комплекса.

3. Комплекс должен обеспечивать выполнение следующих задач:

3.1. Прием первичной сейсмоакустической информации, навигационных параметров, прием и фиксацию момента срабатывания источника упругих колебаний;

3.2. Тестирование секций сейсмокосы;

3.3. Управление секциями стримера.

4. Входными данными могут являться следующие данные:

4.1. Сейсмоакустические данные в сетевом пакете (кадр).

4.2. Навигационные параметры в сетевом пакете.

4.3. Момент времени срабатывания источника упругих колебаний (пушек) в сетевом пакете.

4.4. Квитанции о прохождении команды в секции сейсмоакустической косы.

5. Выходными данными являются следующие данные:

5.1. Сейсмоакустическая информация в бинарном (двоичном) файле.

5.2. Навигационные параметры в бинарном файле.

5.3. Момент (время в микросекундах) срабатывания источника упругих колебаний (пушек) в бинарном файле.

5.4. Команды для управления секциями сейсмоакустической косы.

Сейсмоакустические исследования на сегодняшний день очень стремительно «вливаются» в инженерные изыскания для изучения геологии массива, нарушения угольных пластов, оценки напряжённого состояния геологической среды, а также для изучения физико-технических характеристик горных пород. Отметим, что такие исследования могут входить в обязательную часть геологических изысканий. Необходимость этих исследований отражена в работах, которые составляется геологами на основании технического задания.

В заключение хотелось бы еще раз отметить важность сейсмоакустических исследований при разработке месторождений полезных ископаемых. Они позволяют не только сократить время работы, но и обезопасить процесс добычи. Особую важность приобретают сейсмоакустические исследования на акваториях, где они обеспечивают безопасную разработку шельфов различной глубины.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Калинин А.В., Калинин В.В., Пивоваров Б.Л. Сейсмоакустические исследования на акваториях. – Недра, Москва, 1983 г., 1983 стр., УДК: 850.334

2. Соллогуб В.Б., Чекунов А.В. Строение земной коры и верхней мантии по данным сейсмических исследований. - Издательство "Наукова Думка", Киев, 1977 г., 321 стр., УДК: 550.834.32.

УДК 004.724

Н.С. ЛЕБЕДЕВА, А.Н. САПРЫКИН

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

ПОИСК ОПТИМАЛЬНЫХ БАЗОВЫХ ПАРАМЕТРОВ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ БАЛАНСИРОВКИ СЕТЕВЫХ ПОТОКОВ В ЦОД

Рассматриваются базовые параметры генетического алгоритма, используемого для балансировки сетевых потоков в ЦОД. Приводится экспериментальное обоснование выбора их рекомендуемых значений.

Суть генетического алгоритма состоит в том, чтобы с помощью операторов скрещивания и мутации создавать новые, отвечающие определенным критериям особи, обладающие в каждом последующем поколении лучшими значениями фитнес-функции. Использование генетического оператора мутации позволяет алгоритму выходить из “тупиковых” ситуаций (локальных экстремумов), и поэтому он является одним из наиболее важных операторов алгоритма. Значение параметра мутации влияет на сходимость генетического алгоритма: слишком малое его значение приводит к снижению генетического разнообразия популяции, а слишком большое – сводит работу алгоритма к обычному случайному поиску.

Прямое влияние на скорость работы и сходимость генетического алгоритма также оказывают количество особей в популяции и число обрабатываемых алгоритмом поколений. От данных параметров зависит скорость работы и сходимость генетического алгоритма, однако, на данный момент не существует общепринятых рекомендуемых значений как для первого параметра, так и для второго. В зависимости от поставленной задачи, значение параметра мутации, число особей в популяции и число поколений может варьироваться. Подбор оптимальных значений указанных параметров зависит от размера поискового пространства и вида целевой фитнес-функции.

Определим оптимальные значения параметров генетического алгоритма балансировки нагрузки в сетях центров обработки данных, содержащих от 16 до 32 сетевых узлов. В ходе проведенного исследования с использованием разработанной программы была проведена серия экспериментов. Эксперименты проводились на ЭВМ, обладающей следующими характеристиками: Intel Core i5 2,8 GHz, RAM 8Gb.

Каждый эксперимент включал несколько этапов. На первом этапе вычислялось оптимальное для скорости работы алгоритма значение количества особей в популяции. На основе полученных на первом этапе результатов на втором этапе определялось оптимальное для сходимости алгоритма значение параметра мутации. Рассмотрим ход эксперимента более подробно.

Этап I. Для проведения эксперимента были использованы следующие значения основных параметров генетического алгоритма балансировки сетевой нагрузки в ЦОД: вероятность скрещивания особей – 80%, общее число поколений – 100, вероятность мутации – 5%, а число особей в популяции изменялось для каждой серии экспериментов, начиная от 10 особей в популяции и заканчивая 200. Меньший или больший размер популяции приводил либо к недостижению оптимального результата, либо к нецелесообразному увеличению времени работы алгоритма при отсутствии значимых результатов. Для каждого значения числа особей в популяции была проведена серия из 10 экспериментов.

Таблица 1. Значения фитнес-функции для заданного числа особей в популяции в сетях ЦОД с двухфазной маршрутизацией.

Число особей в популяции	Значение фитнес-функции, условные единицы			Значение фитнес-функции, условные единицы		
	Сеть из 16 узлов			Сеть из 32 узлов		
	Средний результат	Худший результат	Лучший результат	Средний результат	Худший результат	Лучший результат
10	695	762	617	1621	1724	1458
20	585	664	523	1523	1694	1281
30	563	690	467	1449	1536	1359
40	539	630	481	1377	1502	1249
50	511	543	459	1409	1506	1318
60	512	533	463	1350	1435	1275
70	496	539	445	1389	1432	1313
80	510	578	473	1363	1489	1203
90	501	557	462	1358	1415	1297
100	479	519	443	1363	1432	1243
110	486	538	455	1364	1577	1291
120	459	514	425	1425	1587	1317
130	497	539	437	1444	1627	1371
140	481	566	453	1405	1548	1249
150	477	508	447	1378	1510	1269
160	480	532	443	1405	1571	1272
170	457	484	435	1389	1476	1314
180	482	513	461	1379	1469	1288
190	456	512	419	1364	1496	1311
200	478	511	429	1352	1448	1229

В таблице 1 отражены значения фитнес-функции в зависимости от числа особей в популяции и количества узлов в сети. Результаты для каждой сети разделены на 3 раздела, где указаны средние, худшие и лучшие результаты для каждого числа особей в популяции. Так как в алгоритме ставится задача минимизировать фитнес-функцию, то наилучшим результатом являются наименьшие значения.

Из проведенного эксперимента видно, что малый размер популяции (10-20 особей) приводит к худшим результатам, чем большой (100-150). Таким образом, оптимальным значением числа особей в поколении является 100. Последующее увеличение данного значение не приводит к лучшим результатам, но при этом значительно увеличивает время работы алгоритма.

Для удобства восприятия приведенные в таблице 1 числовые результаты были переведены в графическую форму (рисунки 1-2). Из представленных графиков видно, что слишком малый размер популяции (10-20 особей) не дает генетическому алгоритму возможности найти точное решение. Получение околооптимальных результатов становится возможным, начиная с размера популяции в 40 особей.

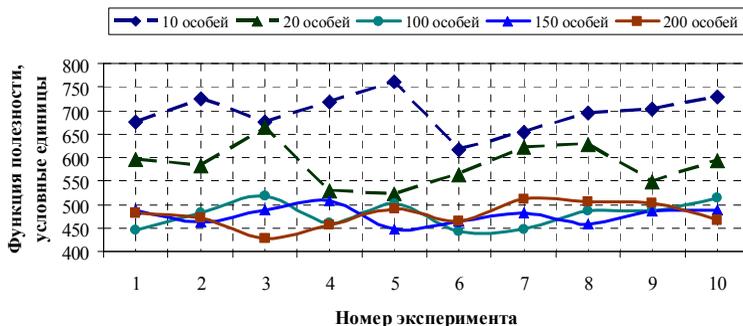


Рисунок 1 – Средние значения фитнес-функции, полученные за серию экспериментов I этапа в сети ЦОД из 16 узлов

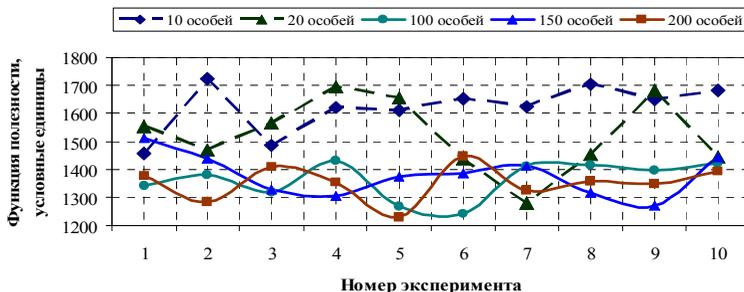


Рисунок 2 – Средние значения фитнес-функции, полученные за серию экспериментов I этапа в сети ЦОД из 32 узлов

По рисункам 1-2 можно сделать вывод, что при малых размерах популяции (10-20 особей) работа алгоритма становится непредсказуемой, значения фитнес-функции на каждой итерации сильно разнятся, и алгоритм не может найти глобальный экстремум. Согласно полученным данным, начиная от численности популяции в 100 особей, последующее увеличение числа особей не оказывало влияния на значения фитнес-функции. Таким образом, можно сделать вывод, что для достижения глобального экстремума и ускорения работы генетического алгоритма балансировки сетевой нагрузки в ЦОД с количеством узлов от 16 до 32 достаточно установить численность популяции на уровне порогового значения в 100 особей.

Проведенная серия экспериментов устанавливала фиксированное число поколений генетического алгоритма, равное 100.

Этап II. Для проведения второго этапа эксперимента были использованы следующие значения основных параметров генетического алгоритма балансировки сетевой нагрузки в ЦОД: вероятность скрещивания особей – 80%, общее число поколений – 100, число особей в популяции – 100, а вероятность мутации изменялась для каждой серии экспериментов, начиная от 1% и заканчивая 50%.

Для каждого значения вероятности мутации вследствие стохастического характера работы генетического алгоритма была проведена серия из 10 экспериментов.

Таблица 2. Значения фитнес-функции для заданного значения вероятности мутации в сетях ЦОД с двухфазной маршрутизацией.

Вероятность мутации, %	Значение фитнес-функции, условные единицы			Значение фитнес-функции, условные единицы		
	Сеть из 16 узлов			Сеть из 32 узлов		
	Средний результат	Худший результат	Лучший результат	Средний результат	Худший результат	Лучший результат
0,01	695	762	617	1621	1724	1458
0,03	585	664	523	1523	1694	1281
0,05	563	690	467	1449	1536	1359
0,07	539	630	481	1377	1502	1249
0,1	511	543	459	1409	1506	1318
0,15	512	533	463	1350	1435	1275
0,2	496	539	445	1389	1432	1313
0,3	626,9	542	726	1435,8	1323	1549
0,4	535,5	479	592	1438,9	1278	1530
0,5	534,2	505	572	1477,4	1372	1622

В таблице 2 отражены значения фитнес-функции в зависимости от заданной вероятности мутации и количества узлов в сети. Результаты для каждой сети разделены на 3 раздела, где указаны средние, худшие и лучшие результаты для каждого значения вероятности мутации. Так как в алгоритме ставится задача минимизировать фитнес-функцию, то наилучшим результатом являются наименьшие значения.

Из проведенного эксперимента видно, что слишком малое значение вероятности мутации (1-4%) и слишком большое значение вероятности мутации (30-50%) приводит к худшим результатам, чем среднее (5-20%).

Для удобства восприятия приведенные в таблице 2 числовые результаты были переведены в графическую форму (рисунок 3). Графики из рисунка 3 описывают изменения значений фитнес-функции в зависимости от варьирования значений вероятности мутации в интервале от 1 до 50% для сети ЦОД с двухфазной маршрутизацией, состоящей из 16 или 32 узлов.

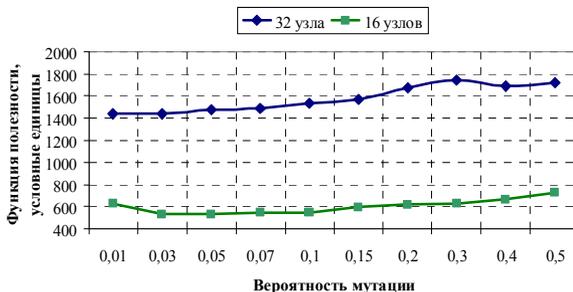


Рисунок 3 – Средние значения фитнес-функции, полученные за серию экспериментов II этапа в сети ЦОД

По рисунку 3 можно сделать вывод, что при непопадании значения вероятности мутации генетического алгоритма в рекомендуемый диапазон от 5 до 10% эффективность предложенного алгоритма снижается на величину до 20%.

Таким образом, можно сделать вывод, что для достижения глобального экстремума и ускорения работы генетического алгоритма балансировки сетевой нагрузки в ЦОД с количеством узлов от 16 до 32 достаточно установить значение вероятности мутации на уровне порогового значения около 5-10%.

В заключение следует отметить, что базовые параметры генетического алгоритма, такие как число особей в поколении, число поколений и вероятность мутации оказывают решающее влияние на время работы и сходимость генетического алгоритма балансировки сетевой нагрузки в ЦОД с двухфазной маршрутизацией. Установка указанных параметров в пределах рекомендуемых значений позволяет получить квазиоптимальное решение, близкое к глобальному оптимуму.

УДК 004.724

Н.С. ЛЕБЕДЕВА, А.Н. САПРЫКИН

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ БАЛАНСИРОВКИ СЕТЕВЫХ ПОТОКОВ В ЦОД

Рассматриваются основные этапы и особенности генетического алгоритма, используемого для балансировки сетевых потоков в ЦОД. Описывается структура хромосомы генетического алгоритма, приводится ее детальный анализ.

В данной статье рассматриваются основные этапы генетического алгоритма балансировки нагрузки в сетях центров обработки данных.

Шаг 1. Формирование стартовой популяции генетического алгоритма. Данный шаг включает в себя создание первоначальной популяции с заданным количеством особей генетического алгоритма, которое осуществляется случайным образом. Используемый принцип случайности применяется для обеспечения стандартного распределения особей алгоритма в поисковом пространстве, что впоследствии приводит к достижению генетического разнообразия.

Популяция алгоритма состоит из набора элементов, которые называются особями. Каждая особь содержит в себе набор закодированных параметров, которые сгруппированы в одну или несколько хромосом. Эти параметры или признаки могут кодироваться несколькими способами: дискретным способом, порядковым способом, древообразным способом или вещественным способом. Выбранный способ кодирования не влияет на итоговое представление хромосомы в программном виде, которое использует либо десятичную, либо двоичную систему счисления. Наиболее распространенным является использование двоичного кодирования хромосом.

В настоящей работе особь рассматривается как совокупность нескольких хромосом, отвечающих за отдельные части сетевого потока. Параметры, включаемые в хромосому, напрямую зависят от способа маршрутизации трафика. В данной работе используются сети с двухфазной маршрутизацией, предполагающие прохождение потока от источника к приемнику через один промежуточный узел и его распараллеливание на две части для снижения нагрузки отдельных каналов, вследствие чего хромосома включает следующие параметры:

коэффициент балансировки и два промежуточных узла для каждой из частей информационного потока.

Коэффициент балансировки является параметром, который определяет процентное соотношение сетевого трафика, проходящего через первый и второй промежуточные узлы. Сетевой трафик, проходящий через первый промежуточный узел, указывается в процентах, в то время как объем трафика для второго узла рассчитывается путем вычитания данного коэффициента балансировки из 100%. В структуре хромосомы данный параметр занимает не более 7 бит. При формировании хромосом возможно появление элементов, содержащих номера узлов, отсутствующих в реальной сети. Такие хромосомы называются запрещенными, и для исключения вероятности их возникновения используется плавающий размер кодирования этого параметра в битах. В зависимости от числа содержащихся в сети узлов этот размер может варьироваться от трех до семи бит. Таким образом, максимальное число сетевых узлов в данной работе может достигать значения 128. Рассмотрим пример кодирования отдельной хромосомы (рисунок 1).



Рисунок 1 – Пример кодирования хромосомы

В данном примере описывается структура хромосомы со следующими параметрами: коэффициенту балансировки присвоено значение (в двоичной системе счисления) 1010000, что соответствует значению 80 в десятичной системе счисления; первому промежуточному узлу присвоено значение 1110000, что соответствует значению 112 в десятичной системе счисления; второму промежуточному узлу присвоено значение 1000110, что соответствует значению 70 в десятичной системе счисления.

Далее рассмотрим, какой вид приобретает особь, состоящая из нескольких хромосом (таблица 1). Каждая хромосома данной особи состоит из 13 бит, 7 из которых отводятся на кодирование коэффициента балансировки, и по 3 бита отводятся на кодирование каждого значения промежуточных узлов.

Таблица 1. Генотип и фенотип особи генетического алгоритма

Генотип	Фенотип		
	Коэффициент балансировки	Промежуточный узел первой части потока	Промежуточный узел второй части потока
0101101001011	45	1	3
0110111011000	55	3	0
1001101010011	77	2	3

В таблице 1 показаны генотип и фенотип особи генетического алгоритма. Генотип представлен 3 хромосомами, закодированными двоичным кодом, а фенотип представляет собой их расшифрованные значения в привычной десятичной системе счисления.

Предложенный способ кодирования обладает рядом преимуществ, например, в структуру хромосомы при необходимости легко можно добавить новые параметры (как коэффициенты балансировки, так и промежуточные узлы сети). Подобная модификация преследует цель увеличить число параллельных сетевых потоков, однако следует принять во внимание, что значительное увеличение длины хромосомы оказывает неблагоприятное влияние на сходимость и быстродействие генетического алгоритма.

Шаг 2. Переконфигурирование ошибочных разделов генотипов. Внутри отдельной хромосомы несколько генов объединяются в группы, в совокупности представляя собой разделы ее генотипа. Каждый раздел описывает конкретный параметр: коэффициент балансировки, номера промежуточных узлов для каждой из частей сетевого потока. Вследствие того, что начальная популяция формируется случайным образом, а число отводимых под определенный параметр бит задается фиксированным образом, иногда в генотипе появляются ошибочные разделы, описывающие большее число узлов, чем существует в сети. Особь, содержащая ошибочный раздел генотипа становится нежизнеспособной. Следовательно, перед началом работы генетический алгоритм должен проверить все особи начальной популяции на присутствие в них ошибочных разделов и их переконфигурировать.

Шаг 3. Расчет значений фитнес-функции. Фитнес-функция является характеристикой полезности особи и вычисляет уровень загрузки сетевых каналов, присваивая повышенные значения штрафной функции для наиболее загруженных. Генетический алгоритм, используемый в данной работе, выделяет в степени загруженности канала заданные интервалы и для каждого из них

использует определенные значения штрафной функции. Наименее загруженные каналы получают минимальный штраф, в то время как наиболее загруженные получают максимальный.

Таким образом фитнес-функция может быть представлена в виде выражения (1).

$$f(x) = \min \left[\sum_{i=1}^n R(c_i(x)) \right], \quad (1)$$

где n – количество сетевых каналов, $c_i(x)$ – значения загруженности i -го канала, $R(c_i(x))$ – значение налагаемого штрафа.

Шаг 4. Отбор особей для новой популяции. Существует 3 основных способа проведения селекции особей: турнирный отбор, элитарный отбор и отбор усечением. Суть турнирного отбора состоит в том, чтобы взять две произвольные особи и сравнить их. Наилучшая из двух получает право участвовать в скрещивании, другая больше не используется. Достоинством данного способа является отсутствие необходимости расчета значений фитнес-функции для всех особей поколения. При отборе по типу элитизма в новое поколение сразу включается особь с наилучшим значением фитнес-функции. Она может быть перенесена в следующее поколение несколько раз и не подвергается скрещиванию и мутации. При отборе усечением производится сортировка особей по значению фитнес-функции, а затем худшие из них отбрасываются. Заданное количество лучших особей переносится непосредственно в новое поколения без скрещивания и мутации. Два последних способа селекции гарантируют, что особи с наиболее высокими значениями фитнес-функции не будут случайно отброшены, как это может произойти при турнирном отборе, и из поколения в поколение уровень полезности не будет снижаться. Так как все поколения обладают фиксированным числом особей, то при использовании двух последних способов селекции оставшееся число особей для нового поколения отбирается с использованием турнирного отбора.

Шаг 5. Скрещивание и мутация отобранных особей. На данном шаге отобранные ранее особи подвергаются сначала скрещиванию, затем мутации (рисунок 2). Генетический оператор скрещивания из двух изначальных особей создает двух новых потомков, унаследовавших гены обоих родителей. Генетический алгоритм, используемый в данной работе, задействует одноточечное скрещивание: обоим изначальным особям в генотипе случайным образом задается точка разрыва, а потомки получают гены, расположенные до точки разрыва, от одной родительской особи, и гены, расположенные после точки разрыва, от другой. После этого

полученные особи-потомки подвергаются мутации. Генетический оператор мутации произвольно изменяет уже имеющиеся в них гены. Количество мутирующих генов может настраиваться. Применение данного генетического оператора помогает алгоритму находить выход из возможных «тупиковых» ситуаций. Следует учитывать, что после применения данных генетических операторов в особях могут появиться ошибочные разделы генотипа, которые далее должны быть подвергнуты переконфигурации.

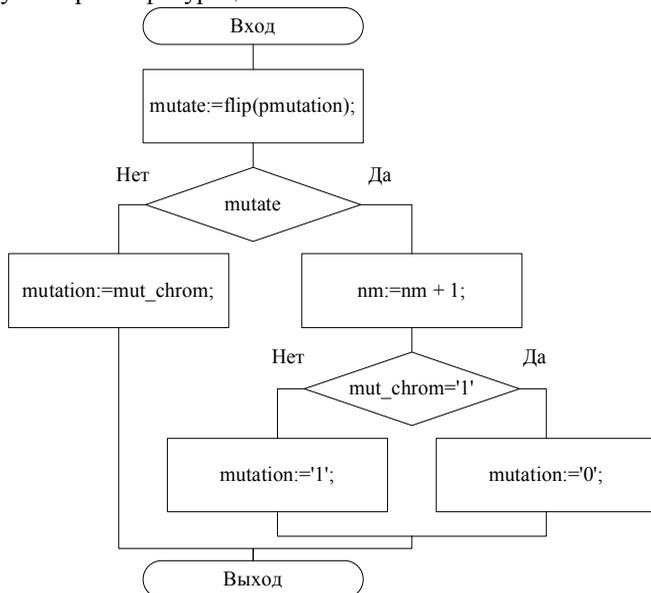


Рисунок 2 – Укрупненная блок-схема процедуры мутации

Шаг 6. Формирование новой популяции. Из потомков, полученных на предыдущем шаге, собирается новое поколение особей.

Шаг 7. Шаги 2-6 повторяются до выполнения одного из следующих условий: достижения заданного для алгоритма числа итераций или получения квазиоптимальных результатов.

УДК 004.896

В.Ю. ЛИКУЧЕВ, Д.А. ПЕРЕПЁЛКИНРязанский государственный радиотехнический университет
имени В.Ф. Уткина**О МНОГОАГЕНТНОМ ПОДХОДЕ К РЕШЕНИЮ ЗАДАЧ
РАЗМЕЩЕНИЯ РАДИОЭЛЕКТРОННЫХ КОМПОНЕНТОВ**

Предложен подход к решению задачи размещения радиоэлектронных компонентов, основанный на имитации поведения самоорганизующейся многоагентной системы. Представлены формализм и обобщенная математическая модель задачи размещения в терминах рассматриваемого подхода.

Размещение радиоэлектронных компонентов (РЭК) на монтажном поле – один из ключевых этапов конструкторского проектирования модулей радиоэлектронной аппаратуры. Оптимальное размещение компонентов обеспечивает повышение надежности аппаратуры, уменьшение размеров конструктивных единиц, минимизацию взаимных наводок, задержек сигналов, уменьшение общей длины соединений и т.п. [1].

На сегодняшний день разработано большое число методов и алгоритмов размещения, различающихся по форме постановки задачи, принципу реализации и структуре построения. Подробную классификацию методов приводит В.М. Курейчик в работе [1].

Однако, несмотря на внушительный массив проведенных исследований, задача размещения РЭК до сих пор не теряет свою актуальность, ввиду увеличения множества конструкционных и физических требований, предъявляемых к аппаратуре. Так как задача размещения относится к классу NP-полных, разработка алгоритма, позволяющего найти оптимальное решение за приемлемое время – весьма затруднительна.

Новым направлением в исследованиях является создание биоинспирированных алгоритмов оптимизации, среди которых выделяют эволюционные и роевые. Первые базируются на использовании механизмов естественной эволюции, таких как наследование, мутация, отбор и кроссинговер (для задачи размещения применены, например, в работах [2, 3]). Вторые основаны на имитации коллективного поведения децентрализованных самоорганизующихся систем, которые состоят из множества простых агентов, локально взаимодействующих между собой и окружающей средой (применены в работе [3, 4]).

Все перечисленные алгоритмы являются итерационными, эвристическими алгоритмами случайного поиска, которые позволяют эффективно решать задачу размещения объектов высокой сложности и нерегулярной структуры (элементы имеют форму прямоугольников кратных размеров либо число разногабаритных элементов не превышает 10% от общего числа элементов [1]), что вполне отвечает условиям проектирования СБИС и печатных узлов с канальной трассировкой электрических связей.

Основной недостаток биоинспирированных методов на данном этапе их развития состоит в том, что они, если можно так выразиться, остаются больше абстрактными, чем объектно-ориентированными, то есть не учитывают или «слабо» учитывают множество параметров размещаемых элементов и множество ограничений, накладываемых на объект размещения в целом, следовательно, не учитывают такие ключевые условия реального процесса проектирования, как:

- *существенная* нерегулярность структуры объекта размещения (элементы имеют форму прямоугольников произвольных размеров либо многоугольников произвольной формы [1]);

- нестандартная конфигурация монтажного поля, наличие запрещенных зон, а также возможность размещения элементов с обратной стороны платы;

- наличие конструктивного базиса, то есть подмножества элементов, положение которых строго определено заранее, например, разъемов;

- наличие подмножеств элементов, взаимное расположение которых критично;

- многокритериальность задачи размещения (критерии простоты последующей трассировки соединений, электромагнитной совместимости, тепловой защищенности и др.), при условии, что большинство критериев выражаются в нечетких переменных-признаках.

Перечисленные условия одновременно имеют место в задачах размещения РЭК при проектировании модулей прецизионной аппаратуры. Вследствие того, что рассматриваемые методы не учитывают все эти условия, разработчики устройств вынуждены размещать компоненты вручную, а не с помощью автоматизированных систем, в которых реализованы подобные методы. Очевидно, что для расширения области применения биоинспирированных алгоритмов, необходимо провести работы по их модификации и интеграции.

Одним из возможных решений проблемы является изменение подхода к представлению объектов задачи, что позволило бы

учитывать расширенное множество связей и отношений между ними. В качестве такого подхода предлагается усовершенствованный многоагентный подход, где в отличие от роевой концепции, агент (в нашем случае РЭК) представляется не только формально, как элемент множества, а как программная сущность с набором признаков и допустимых операций с ним, а сама система (объект размещения) – не как рой примитивных частиц, а как коллектив интеллектуальных агентов с более выраженным признаком социальности.

Для решения поставленной задачи формализуем в первом приближении модель многоагентной системы и задачу размещения для нее. Пусть $A = \{a_i | i = 1, 2, \dots, n\}$ – множество агентов системы (множество размещаемых РЭК), а вектор $P = (p_{10}, p_{20}, \dots, p_{n0})$ – набор параметров каждого агента, который фактически характеризует состояние последнего и отражает всю информацию о нем. В данном случае $P = (x_i, y_i, w_i, h_i, p_{10}, p_{20}, \dots, p_{n0})$, где x_i и y_i – координаты геометрического центра РЭК, w_i и h_i – габариты РЭК, $p_{10}, p_{20}, \dots, p_{n0}$ – другие параметры, например, признаки ориентации и расположения РЭК на конкретной стороне платы; приведенные к интервалу $[0, 1]$ оценки мощности рассеивания и уровня электромагнитных помех.

Агенты в коллективе находятся во взаимных связях и отношениях. В нашей задаче РЭК связаны между собой в соответствии со схемой электрической принципиальной. Такая система моделируется гиперграфом $H = (Q, E)$, где $Q = \{q_i | 1, 2, \dots, n\}$ – множество вершин, моделирующих элементы, $E = \{e_j | 1, 2, \dots, m\}$ – множество гиперребер, моделирующих электрические цепи. Агенту ставится в соответствие изменяющийся во времени вектор связности, который показывает, с какими агентами находится в отношениях рассматриваемый агент в данный момент по одному из параметров, а также величины этих отношений. Для системы можно записать матрицу связности:

$$R = \begin{matrix} & a_1 & \dots & a_n \\ \begin{matrix} a_1 \\ \vdots \\ a_n \end{matrix} & \begin{bmatrix} r_{11} & \dots & r_{1n} \\ \vdots & \ddots & \vdots \\ r_{n1} & \dots & r_{nn} \end{bmatrix} \end{matrix},$$

где r_{ij} – величина отношения двух агентов, которая может выражаться однозначно либо нечеткой переменной. Для системы размещаемых РЭК в первую очередь строится матрица связности по расположению, то есть по параметрам x_i, y_i . Для примера рассмотрим матрицу трех РЭК, располагающихся на одной линии:

$$R = \begin{matrix} & \alpha_1 & \alpha_2 & \alpha_3 \\ \alpha_1 & \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \end{matrix}.$$

Положим, если $r_{ij} = 0$ – то два элемента расположены не рядом (в соответствии с некоторой оценкой взаимной удаленности геометрических центров), а если $r_{ij} = 1$, то они расположены рядом друг с другом. По матрице видно, что компонент α_3 расположен посередине.

Каждый агент способен выполнять набор элементарных операций в соответствии с заданным набором правил. Выполнение операции состоит в изменении одного или нескольких параметров вектора P . Можно говорить о параметрической алгебре агентов, так как существует множество параметров и заданные на этом множестве операции. В задаче размещения на множестве $\{x_i, y_i | i = 1, 2, \dots, n\}$ определена операция линейного перемещения $+(\Delta x_{it}, \Delta y_{it}) : (x_i, y_i) \rightarrow (x_i + \Delta x_{it}, y_i + \Delta y_{it})$, где Δx_{it} и Δy_{it} – приращения координат в данный момент времени. Выполнение операции производится согласно правилам, которые учитывают вектор P агента, гиперграф H , вид соответствующего вектора связности матрицы R , векторы P агентов, находящихся в отношениях с рассматриваемым агентом и условия окрестности среды (линейные ограничения на плоскости) в данный момент. Говоря простым языком, агент «ощущает» свое состояние и положение относительно связанных агентов, и анализируя данные, принимает решение о выполнении той или иной операции. Это является одним из ключевых принципов поведения многоагентных систем, а также говорит о детерминированном характере поиска агентами своих оптимальных состояний.

Учитывая положения выше, можно рассматривать P как изменяющийся вектор, который включает в себя *постоянные* и *изменяемые* по величине; изменяемые по величине *зависимо* и *независимо* параметры, причем:

- один и тот же параметр разных агентов может быть постоянным и изменяемым по величине; изменяемым *зависимо* и *независимо*;
- параметр агента может быть изменяемым *зависимо* от времени или от величины одного, или нескольких параметров другого агента (агентов).

Для примера, в векторе $P = (x, y, w, h, p_1, p_2, \dots, p_k)$ параметры w, h у всех РЭК постоянны, а x, y – у одних РЭК могут быть постоянными (такие РЭК входят в конструктивный базис), а у других могут изменять свою величину в зависимости от величины x, y связанных РЭК при выполнении операции перемещения.

Являясь элементом системы, агент не может оценивать состояние всей системы в процессе решения некоторой задачи, и в него не заложено полное представление о целевой функции оптимизации системы. Располагая лишь данными о связанных агентах и окрестности пространства, каждый агент путем выполнения операций стремится оптимизировать лишь свою собственную, частную целевую функцию (причем поиск осуществляется лишь в окрестности агента), а система оптимизирует частные целевые функции всех агентов, то есть ее задача всегда – многокритериальная.

Частная (агентная) задача оптимизации для i -го агента имеет

вид: $\min_{P_i \in D_i} (w_i, \vec{f}_i(\vec{P}_i))$, где \vec{P}_i – множество решений вектора параметров, относящиеся к области всех возможных решений D_i ; $\vec{f}_i(\vec{P}_i)$ – множество критериев; w_i – множество весовых коэффициентов, а задача для всей системы:

$\min_{P_i \in D_i} F = (g_1, g_2, \dots, g_n, f_1(P_1), f_2(P_2), \dots, f_n(P_n))$. Возможное различие весовых коэффициентов в задачах означает, что в общем случае какой-либо более значимый критерий для агента, может быть менее значимым для всей системы и наоборот.

Рассмотрим случай с размещением РЭК. В качестве критериев оптимизации выберем нормированные оценки общей длины связей $O_L(f(\vec{C}_L))$ и занимаемой площади $O_S(f(\vec{C}_S))$ (для одного РЭК имеет только одно решение), где $\vec{C}_L, \vec{C}_S \in \vec{P}_i$, так как критерии зависят не от всех параметров. Задача оптимизации для одного РЭК заключается в отыскании такого положения на монтажном поле, при котором общая длина его связей была бы минимальной, то есть формально:

$\min_{P_i \in D_i} (w_L, w_S, O_L(f(\vec{C}_L)), O_S(f(\vec{C}_S)))$. Задача для всего объекта состоит в отыскании оптимальных положений всех РЭК, при которых длина всех связей и общая занимаемая площадь были бы минимальными:

$$\min_{F_i \in Z_F} F = ((y_L, y_S, O_L(f(\overline{C}_{L_i})), O_S(f(\overline{C}_{S_i}))) | i = 1, 2, \dots, n)$$

. Отметим, что $w_S = 0$, и $O_S(f(\overline{C}_{S_i}))$ может быть опущена в задаче для одного компонента, однако, в глобальной задаче при составлении аддитивного критерия эта оценка приобретает весомый смысл.

Безусловно, описанная модель задачи размещения РЭК и метода ее решения требует анализа и дополнения. Наиболее интересные вопросы касаются динамики РЭК, то есть правил, по которым происходит перемещение (вид этих правил и зависимость их числа от числа размещаемых компонентов), и скорости перемещения (для всех ли компонентов она будет одинаковой?). Аналогия с социальной системой ставит вопрос о введении в модель составляющей взаимодействия РЭК не только посредством взаимной ориентации, но и путем обмена данными о среде. Сюда же можно отнести вопрос о целесообразности наделения компонентов ментальными свойствами, наличие которых предполагается принципами поведения многоагентных систем.

Однако, уже сейчас видно, что приведенная модель может давать представление об объекте размещения во всей полноте его свойств, а представление о задаче – во всей полноте ее требований и условий. При верной реализации в виде алгоритмов (с возможностью отсека лишних поисковых операций), и, далее, в виде автоматизированной системы, рассмотренная модель позволит эффективно решать задачи размещения РЭК при проектировании модулей прецизионной аппаратуры, отвечая всем соответствующим требованиям.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Курейчик В.М. Математическое обеспечение конструкторского и технологического проектирования с применением САПР: Учебник для вузов. – М.: Радио и связь, 1990. – 352с.: ил.
2. Бушин С.А., В.В. Курейчик В.В. Генетический алгоритм размещения разногабаритных элементов// Известия ЮФУ. Технические науки. – 2009. – № 12 (101). – С. 22-27.
3. Лебедев Б.К., Лебедев О.Б., Лебедев В.Б. Гибридизация роевого интеллекта и генетической эволюции на примере размещения// Программные продукты, системы и алгоритмы. – 2017. – №4.
4. Лебедев Б.К., Лебедев В.Б. Размещение на основе метода пчелиной колонии// Известия ЮФУ. Технические науки. – 2010. – № 12 (113). – С. 12-19.

5. Ю.Л. Муромцев, Д. Ю. Муромцев, И. В.Тюрин и др. Информационные технологии проектирования радиоэлектронных средств учебное пособие для студентов высших учебных заведений — М. Издательский центр «Академия», 2010. — 384 с.

УДК 004.942

Н.А. МАРКОВА

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

ОБЗОР И ИССЛЕДОВАНИЕ МИКРОКОНТРОЛЛЕРОВ

Рассматриваются современные варианты микроконтроллеров исходя из их характеристик.

Микроконтроллер (MS, англ. Микроконтроллер (MCU) - это чип, который объединяет функции процессора и периферийных устройств на одном чипе, содержащем ОЗУ или же ПЗУ. Другими словами - это особая микросхема, разработанная для управления различными электронными устройствами. Микроконтроллеры впервые были разработаны в том же году, что и микропроцессоры общего назначения (1971). В 1979 году в СССР научно-исследовательский институт ТТ разработал 16-битный одночиповый компьютер К1801ВЭ1, микроархитектура которого была названа "электроника НК". В СССР проводилась разработка уникальных микроконтроллеров, а также создание более удачных за рубежом клонов образцов (например, CR1816V51-популярный аналог и т.д.

В отличие от персональных компьютеров, которые считаются универсальными высокопроизводительными микропроцессорами, микроконтроллеры применяются для интеграции во всевозможные приборы. При разработке микроконтроллеров существует необходимость поддерживать баланс между размером и ценой, с одной стороны, и гибкостью, и производительностью, с иной. Подходящее соотношение этих и иных характеристик для всевозможных приложений может очень сильно различаться. В отличие от типичных микропроцессоров, микроконтроллеры нередко используют архитектуру памяти Гарварда, то есть соответственно отдельное хранение данных и команд в ОЗУ и ПЗУ.

С выходом в свет одночиповых микроЭВМ связано начало времени глобального использования компьютерной автоматизации в области управления. В связи с понижением внутреннего изготовления и наращиванием импортного оснащения, в том числе компьютерной техники, термин "микроконтроллер" (МК) вытеснил ранее применяемый термин «одночиповые микроЭВМ».

Неполный перечень периферийных приборов, которые могут присутствовать в микроконтроллерах, включает:

- универсальные цифровые порты (они могут быть настроены как на вход, так и на выход);
- различные интерфейсы ввода-вывода, например: UART, I²C, SPI, CAN, USB, IEEE 1394, Ethernet;
- компараторы;
- аналого-цифровые и цифроаналоговые преобразователи;
- широтно-импульсные модуляторы;
- контроллеры бесколлекторных двигателей;
- таймеры;
- контроллеры дисплеев и клавиатур;
- массивы встроенной флеш-памяти;
- радиочастотные приемники и передатчики;
- встроенный тактовый генератор и сторожевой таймер.

На сегодняшний день наиболее известные семейства микроконтроллеров:

- MCS 51 (Intel)
- MSP430 (TI)
- ARM (ARM Limited)
- AVR (Atmel) (ATmega, ATtiny, XMega)
- PIC (Microchip).

Классификация микроконтроллеров.

Микроконтроллеры характеризуются в основном шириной шины, набором команд, а также структурой памяти. Для одной и той же семьи могут существовать разные формы с разными источниками. Классификация по количеству бит:

1) 8-битный микроконтроллер: процессор или ALU обрабатывают 8-битные данные одновременно. Образцами являются 8-битные микроконтроллеры Intel 8031/8051. Они используются в позиционных системах управления, системах управления скоростью [2].

2) 16-битный микроконтроллер: гарантирует более высокую точность и производительность по сравнению с 8-битным. Он предусмотрен для скоростных приложений, таких как робототехника и т. д. Некоторые примеры 16-битных микроконтроллеров - это 16-битные микроконтроллеры, которые расширяют семейства Intel 8096 и Motorola MC68HC12 [2].

3) 32-битный микроконтроллер: пользуется 32-битными инструкциями по выполнению арифметических и логических операций. Он предназначен для высокоскоростных приложений в

обработке изображений, телекоммуникациях, интеллектуальных системах управления и т. д. [3]. В качестве примера могут выступать: Intel / Atmel 251, PIC 3X, семейство ARM.

Типы микроконтроллеров.

1. MCS 51 - это восьмибитный микроконтроллер, произведенный в 1981 году Intel. Его можно найти в 40-контактном погружении, то есть в линейном корпусе. Он является основным (классическим) микроконтроллером, но до сегодняшнего дня многие компании производят такой тип микроконтроллера. Старые типы 8051 имеют 12 тактовых циклов на одну инструкцию, что делает их медленными, в то время как последние 8051 имеют 6 тактовых циклов на одну инструкцию. Микроконтроллер 8051 не имеет встроенной памяти 33 и A / D-преобразователей, а такие микроконтроллеры-процессоры CISC, а 8051 использует архитектуру фон Неймана [3].

2. MSP430-это первая 16-разрядная модель RISC-микроконтроллера Texas Instruments, представленная в 1999 году. Дополнение для клиента (для тех времен) количество бит, представители этой семьи отличало также и низким потреблением энергии, что сразу же привлекло внимание разработчиков устройств с батарейным питанием, в которых, как известно, сокращение текущего потребления даже на один мкА позволяет значительно увеличить срок службы батареи. Сегодня семейство MSP430 включает более 500 устройств и с каждым годом набирает все большую популярность среди разработчиков электроники благодаря ряду ключевых преимуществ [2].

3. ARM (ARM Limited) разработаны на собственной архитектуре. В семейство входят 32-х и 64-битовые МК. ARM Limited занимается лишь только разработкой ядер и их инструментов, а другие компании могут приобрести лицензии на производство. Данные процессоры потребляют незначительное количество энергии, в следствие этого находят обширное использование в производстве мобильных телефонов, игровых консолей, маршрутизаторов и т. д. К фирмам, которые купили лицензии, относятся: STMicroelectronics, Samsung, Sony Ericsson и др.

4. AVR - семейство восьмибитных микроконтроллеров Atmel, впервые выпущенных в 1996 году. Они являются мощным инструментом, универсальной основой для создания современных и удобных многоцелевых встроенных систем. Идея разработки нового ядра RISC принадлежит двум студентам Норвежского университета науки и техники (Тронхейм) – Alfa Bogen (Alf-Egil Bogen) и Vegard wollen (Vegard Wollen). В 1995 году Боген и Воллен решили

предложить американской компании Atmel создать новый 8-битный микроконтроллер RISC и оснастить его флэш-памятью для программ на одном чипе с вычислительным ядром. Микроконтроллеры AVR имеют архитектуру Гарварда (программа и данные находятся в разных адресных пространствах) и командную систему, близкую к идеологии RISC. Процессор AVR имеет 32 общих 8-битных регистра, объединенных в один файл журнала. В отличие от "идеального" процессора RISC, регистры абсолютно не одинаковы: - три "двойных" 16-битных указателя регистров X (от R26: P27 до Z. R28: R29) и Z (по R30: r31); - некоторые команды работают только с регистрами R16...r31; - результат умножения (в тех моделях, что есть модуль умножения) всегда помещается в r0: r1.

5. PIC Microcontroller (Контроллер периферийного интерфейса) – это семейство микроконтроллеров, основанных на технологии микрочипов США с архитектурой Harvard. Первоначально он был разработан как вспомогательное устройство для программного обеспечения для передачи данных (PDP) для поддержки своих периферийных устройств, а затем получил название PIC. Микроконтроллеры pico являются процессорами RISC. Самое интересное в PIC заключается в том, что его машинный цикл состоит всего из 4 тактовых импульсов, в отличие от 12 тактовых импульсов в микроконтроллере Intel 8051. Микроконтроллеры PIC находят свое применение в новых приложениях, таких как смартфоны, аудио-аксессуары, периферийные устройства для видеогр и передовые медицинские устройства [1].

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Microcontroller Types and Application. [Электронный ресурс]. Режим доступа: <http://www.electronicshub.org/microcontrollers>. - Дата доступа: 01.02.2021.
2. Решения на базе микроконтроллеров MSP430. [Электронный ресурс]. Режим доступа: <https://www.compel.ru/lib/132890>. - Дата доступа: 01.02.2021.
3. В.Н. Веприк, В.А. Афанасьев, А.И. Дружинин, А.А. Земсков, А.Р. Исаев, О.В. Малявко. Микроконтроллеры семейства MCS-51: Учебное пособие. - Новосибирск.-1997.-62 с.

УДК 004.9

А.Д. МАСЛОВА, А.Р. ПЛАКСИНА
Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

КОНЦЕПТУАЛЬНЫЕ ОСНОВЫ МОБИЛЬНОГО ОБУЧЕНИЯ СТУДЕНТОВ

Рассматриваются возможности использования мобильных технологий в образовательных процессах.

В высшем образовании мобильные устройства используются в качестве жизненно важного компонента образовательных технологий. Мобильные устройства в образовании позволяют студентам получать знания, работать в партнерстве и вносить свой вклад в мысли друг друга. Таким образом, цель этой работы состоит в том, чтобы исследовать и признать жизненно важные факты, влияющие на поведенческое намерение пользователей использовать м-обучение.

Мобильное обучение позволяет дать студентам дополнительную и интересную учебную поддержку, адаптивность к получению новой информации, более широкий канал корреляции. Для характеристики данной системы студентами и сотрудниками Технического университета Пакистана используются термин «web-based social networking» в противоположность несколько устаревшему «web 2.0». Разработанные ими устройства позволяют студентам общаться и работать вместе друг с другом, а также с преподавателями.

Для расширения единой теории принятия и использования технологии (ЕТПИТ) для мобильных устройств в образовании с добавлением переменных интенционального использования самоуправления обучением проведем анализ мобильного обучения.

Теория ЕТПИТ использует систему степеней, а также более позднюю практику выполнения. Модель ЕТПИТ обладает переменными, представляющими собой ожидаемую производительность, нагрузку и социальное влияние, а также детерминацию мотива использования системы данных и эффективности использования, как показано на рисунке 1.

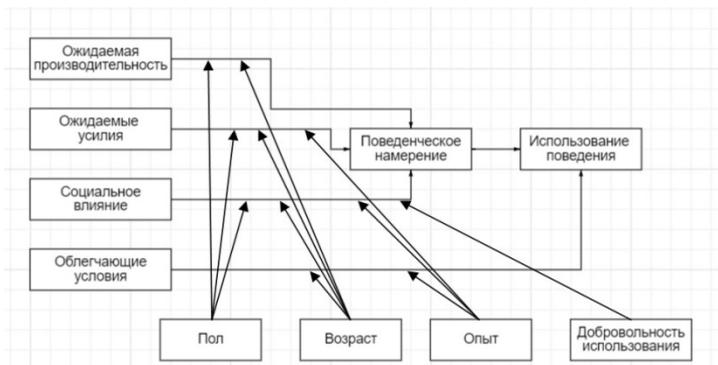


Рисунок 1 – Модель ЕТПИТ

В Техническом университете Пакистана сообщили, что добровольность использования, возраст, пол и опыт обосновывают влияние четырех ключевых переменных на мотивацию использования и производительность.

Исследователи модели ЕТПИТ провели ассоциированную экспериментальную работу по изучению восьми конкурентоспособных моделей IS. ЕТПИТ строит и связывает теории, чтобы обратить внимание на границы более раннего анализа и гипотезы. Кандидаты были из ассоциации, а не из мира. Размеры были выбраны в начале технологии, как и другие уровни деятельности заявителя с этой технологией. Вид был сформирован с использованием пятибалльной шкалы Лайкерта. Выбранная переменная вычисляет ранее действительные вещи, которые должны быть использованы в технологическом и структурном анализе.

Далее была разработана следующая концептуальная структура и гипотеза развития:

1. Интенция поведения – это степень, в которой индивид намеревается принять или приобрести технологию, систему или продукт в будущем. Перспектива мобильной обучающей информационной системы предназначена для измерения или прогнозирования возможных действий специалиста в области образования.

2. Самоэффективность – это способности человека организовывать и выполнять действия, необходимые для достижения заданных результатов. Это понятие было использовано для изучения манер, поведения и намерений человека в различных областях.

3. Воспринимаемая полезность - это степень, в которой человек верит, что использование конкретной технологии/услуги повысит его

или ее производительность. Основной переменной воспринимаемая полезность.

4. Воспринимаемая простота использования - это степень, в которой человек считает, что использование конкретной системы будет свободным от усилий.

5. Ожидаемая производительность - это показатель, рассчитываемый при использовании информационной системы.

6. Социальное влияние – это степень, в которой индивид уверен, что он должен использовать новую систему.

Мобильное обучение быстро развивается и растет. С другой стороны, существуют значительные ограничения в работе с учебными информационными процедурами смартфонов и использованием мобильных устройств в ВУЗах. Данное исследование требует дополнительного изучения, но уже сейчас преподаватели университетов могут дополнительно использовать эти факты в качестве основы для применения ИТ.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Шарпс М. Теория изучения мобильной эры. 2010. 87-99 с.
2. Голицына И. Н., Половникова Н. Л. Мобильное обучение как новая технология в образовании. 2009.
3. M-learning в современном образовательном процессе: За и против [Электронный ресурс].

УДК 004.023

И.А. МУЗЮКИН, А.Н. САПРЫКИНРязанский государственный радиотехнический университет
имени В.Ф. Уткина**МОДИФИЦИРОВАННЫЙ ОПЕРАТОР МИГРАЦИИ В
ОСТРОВНОЙ МОДЕЛИ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ
КОМПОНОВКИ БЛОКОВ ЭЛЕКТРОННЫХ СРЕДСТВ**

Рассматривается применение островного генетического алгоритма с модифицированным оператором миграции.

Конструкторское проектирование является одним из важных этапов автоматизированного проектирования цифровой электронной аппаратуры. Задачи, которые решаются на данном этапе, очень часто бывают многокритериальными, а решение такого типа задач может быть получено только в результате полного перебора.

Генетические алгоритмы (ГА) – это эффективный инструмент, который используется для решения сложных задач, нашедших применение в сферах оптимизации, искусственного интеллекта, техники и в других областях [1].

Островная модель является наиболее распространённой моделью параллельного генетического алгоритма. Ее суть заключается в том, что популяция, состоящая из огромного числа особей, разбивается на одинаковые подпопуляции – острова (islands). Каждый из этих островов обрабатывается отдельно, с помощью классических генетических операторов: скрещивания, мутации и селекции. Эти подпопуляции развиваются отдельно и независимо друг от друга в течение одинакового количества поколений T (время изоляции). По истечению времени изоляции происходит обмен лучшими особями между островами. Данный процесс получил название – *миграция* (рисунок 1).

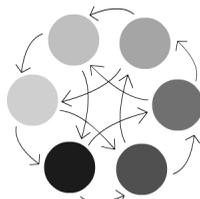


Рисунок 1 – Схема миграции в островной модели генетического алгоритма

Введение миграции в островную модель позволяет находить различные особи – доминанты в подпопуляциях, что способствует поддержанию многообразия. Однако стоит отметить, что очень важно установить правильную частоту миграции:

- чересчур частая миграция (или миграция слишком большого числа особей) приведет к смешению всех подпопуляций, и тогда островная модель будет несильно отличаться от обычного генетического алгоритма [2].
- если миграция будет слишком редкой, то она не сможет предотвратить преждевременное схождение подпопуляций [2].

Работу модифицированного оператора миграции рассмотрим на примере:

Требуется скомпоновать 6 элементов по двум блокам с минимизацией количества межблочных связей. В результате работы генетического алгоритма получим 2 острова, на которых происходит процесс эволюции. Один раз в заданное количество поколений происходит миграция (копирование) лучшей особи с одного острова на другой. Перед миграцией с помощью целевой функции оценим количество межблочных связей. Так как острова в процессе эволюции могут прийти к похожим решениям, но с разной индексацией блоков. Чтобы особь одного острова при миграции была совместима с особями другого острова, применим *механизм реиндексации блоков*.

остров1:

особь1{4}:

блок1: [1, 2, 3]

блок2: [4, 5, 6]

особь2{6}:

блок1: [1, 2, 4]

блок2: [3, 5, 6]

остров2:

особьN{5}:

блокX: [4, 5, 6]

блокY: [1, 2, 3] где X и Y - номера блоков

Определим степень родства особей внутри островов. На схеме видно, что *особь1* на первом острове и *особьN* на втором острове максимально похожи, за исключением того, что блоки у них поменяны местами и значение целевой функции у первой особи лучше, чем у

второй. Поэтому для улучшения решения можем реиндексировать (переименовать) блоки у особи N на втором острове. Блок X стал блоком 2 , а блок Y стал блоком 1 . В результате получим: особь N улучшила показатель целевой функции и стала более пригодной для миграции на другой остров.

остров1:

особь $1\{4\}$:

блок1: [1, 2, 3]

блок2: [4, 5, 6]

особь $2\{6\}$:

блок1: [1, 2, 4]

блок2: [3, 5, 6]

....

особь $N\{4\}$:

блок2: [4, 5, 6]

блок1: [1, 2, 3]

Данный подход позволит:

1. Получать более жизнеспособные особи при скрещивании мигрировавших особей с другими особями острова.
2. Избегать формирования изолированных групп путем мутации от мигрировавших особей, не способных давать потомство при скрещивании с особями из основной группы острова.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Эррера Ф., Лозано М., Вердегай Дж. Л.. Решение real-coded генетических алгоритмов: операторы и инструменты для анализа поведения // Vol. 12, № 4, 1998. – с. 265-319.
2. Интеллектуальные информационные системы : учебное пособие / А. А. Смагин, С. В. Липатова, А. С. Мельниченко. – Ульяновск : УлГУ, 2010. – 136 с.

УДК 004.9

Я.И. НОВИКОВРязанский государственный радиотехнический университет
имени В.Ф. Уткина**ИНТЕРПОЛИРОВАНИЕ ФУНКЦИЙ И ПОСТРОЕНИЕ
ИНТЕРПОЛЯЦИОННЫХ МНОГОЧЛЕНОВ**

Рассматриваются теоретические основы интерполирования функций. Также рассматривается возможность построение интерполяционных многочленов с помощью программных средств.

Что такое интерполирование функций? Какова задача интерполирования? В данной статье мы постараемся ответить на эти вопросы, а также рассмотрим построение интерполяционных многочленов с помощью программных средств, а именно с помощью реализации программы на языке Python.

Во-первых, следует точно понимать, что такое интерполяция, и для чего она используется.

Интерполяция – это способ нахождения промежуточных значений величины по имеющемуся дискретному набору известных значений.

В научных и инженерных расчетах часто приходится оперировать набором значений, полученных экспериментальным путем или методом случайной выборки. Как правило, на основании этих наборов требуется построить функцию, на которую могли бы с высокой точностью попадать другие получаемые значения [1].

Сформулируем основную задачу интерполирования: на отрезке $[a, b]$ заданы $n + 1$ точки x_0, x_1, \dots, x_n , которые называются узлами интерполяции, и значения некоторой функции $f(x)$ в этих точках $f(x_0)=y_0, f(x_1)=y_1, \dots, f(x_n)=y_n$.

Требуется построить функцию $F(x)$ (интерполирующая функция), принимающую в узлах интерполяции те же значения, что и $f(x)$, т.е. такую, что $F(x_0)=y_0, F(x_1)=y_1, \dots, F(x_n)=y_n$.

То есть необходимо найти кривую $y = F(x)$ некоторого определенного типа, проходящую через заданную систему точек $M_i(x_i, y_i), (i=1, 2, \dots)$ (рисунок 1).

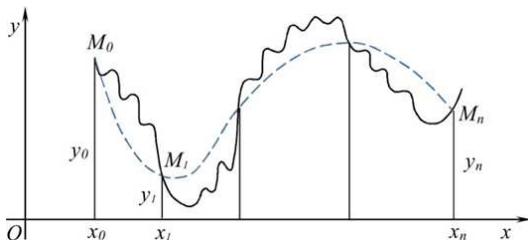


Рисунок 1 – Требуемая кривая

Главной проблемой в такой общей постановке является то, что в худшем случае задача может совсем не иметь решений, а в лучшем – иметь бесчисленное множество этих решений.

Для решения этой проблемы следует вместо произвольной функции $F(x)$ искать полином $P_n(x)$ степени не выше n , т.е. такой, что $P_n(x_0)=y_0, P_n(x_1)=y_1, \dots, P_n(x_n)=y_n$.

Полученную интерполяционную формулу $y = F(x)$ обычно используют для приближенного вычисления значений данной функции $f(x)$ для значений аргумента x , отличных от узлов интерполирования. Такая операция называется интерполированием функции $f(x)$ [1].

Иногда встречаются задачи, когда приходится рассматривать функции $y = f(x)$, значения $y_i = f(x_i)$ для системы равноотстоящих точек x_i ($i=0,1,2, \dots; \Delta x_i = \text{const}$) которых заданы таблично.

Для этого используются конечные разности различных порядков. Разности удобно располагать в форме таблиц двух видов: горизонтальной таблицы разностей или диагональной таблицы разностей (таблица 1, 2).

Таблица 1. Горизонтальная таблица разностей.

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$
x_0	y_0	Δy_0	$\Delta^2 y_0$	$\Delta^3 y_0$
x_1	y_1	Δy_1	$\Delta^2 y_1$	$\Delta^3 y_1$
x_2	y_2	Δy_2	$\Delta^2 y_2$	$\Delta^3 y_2$
\dots	\dots	\dots	\dots	\dots
\dots	\dots	\dots	\dots	\dots

Таблица 2. Вертикальная таблица разностей.

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$
x_0	y_0	Δy_0	$\Delta^2 y_0$	$\Delta^3 y_0$
x_1	y_1	Δy_1	$\Delta^2 y_1$	
x_2	y_2	Δy_2		
x_3	y_3			

Интерполирование функций осуществляется с помощью интерполяционной формулы Ньютона с использованием разделенных разностей [2].

$$L(x) = f(x_0) + f(x_0; x_1)(x-x_0) + f(x_0; x_1; x_2)(x-x_0)(x-x_1) + \dots \\ \dots + f(x_0; \dots; x_n)(x-x_0)\dots(x-x_{n-1}) \quad (1)$$

$$L(x) = f(x_n) + f(x_{n-1}; x_n)(x-x_n) + f(x_{n-2}; x_{n-1}; x_n)(x-x_n)(x-x_{n-1}) + \dots \\ \dots + f(x_0; \dots; x_n)(x-x_n)\dots(x-x_1) \quad (2)$$

Формулу (1) называют формулой для интерполирования вперед, а формулу (2) – формулой для интерполирования назад.

Интерполирование функций нередко используется для решения задач вычислительной математики. Чтобы автоматизировать рутинные и однотипные действия при нахождении полинома $P_n(x)$ n -ной степени, напишем программу построения интерполяционных многочленов с помощью формулы (2) для функции

$$f(x) = \frac{1}{1 + 25x^2}, x_{\min} = -1, x_{\max} = 1, n = 6$$

Разработка будет осуществляться в среде PyCharm. В качестве языка реализации выбран Python [3].

Код программы:

```
import numpy
import matplotlib
import matplotlib.pyplot as plt
xmin=-1 xmax=1
h = (xmin-xmax)/6
def F(x):
    return 1/(1+25*x*x)
def f0(x0):
    return F(x0)
def f1(x0, x1):
    return (F(x1) - F(x0))/(x1 - x0)
def f2(x0, x1, x2):
    return (f1(x1, x2)-f1(x0, x1))/(x2-x0)
def f3(x0, x1, x2, x3):
    return (f2(x1, x2, x3)-f2(x0, x1, x2))/(x3-x0)
def f4(x0, x1, x2, x3, x4):
    return (f3(x1, x2, x3, x4)-f3(x0, x1, x2, x3))/(x4-
x0)
def f5(x0, x1, x2, x3, x4, x5):
    return (f4(x1, x2, x3, x4, x5)-f4(x0, x1, x2, x3,
x4))/(x5-x0)
def f6(x0, x1, x2, x3, x4, x5, x6):
    return (f5(x1, x2, x3, x4, x5, x6)-f5(x0, x1, x2,
x3, x4, x5))/(x6-x0)
def interpol(x, x0, x1, x2, x3, x4, x5, x6):
```

```

#Расчет функции
x = numpy.linspace(-1, 1, 150)
y = F(x)
y_interpol = interpol(x, xmax, xmax - h, xmax - 2 *
h, xmax - 3 * h, xmax - 4 * h, xmax - 5 * h, xmin)
error = y - y_interpol #погрешность
fig, ax = plt.subplots(figsize=(10,6)) #определяем
одну фигуру fig, содержащую один объект Axes - ax
fig.canvas.set_window_title('') #заголовок фигуры
ax.set_title('Построение интерполяционных
многочленов') #заголовок для ax
ax.set_xlim([-1,1])# диапазон значений x
ax.set_ylim([-1.5,1.5])# диапазон значений y
ax.set_xlabel('x')# подпись оси x ax.set_ylabel('F(x)')#
подпись оси y
ax.plot(x, y, color="blue", label="F(x) =
1/(1+25*x*x)", linewidth = 2 )
ax.plot(x, y_interpol, 'b--', color="green",
label="Интерполяционная ф-я", linewidth = 2) ax.plot(x,
error, color="red", label="Погрешность", linewidth = 1 )
ax.legend()# добавление легенды
plt.show()

```

Результат работы программы представлен на рисунке 2:

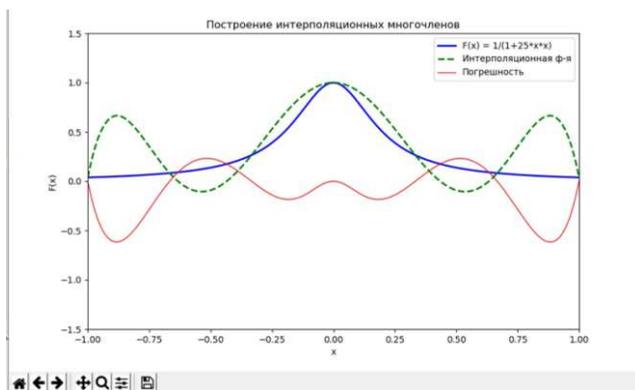


Рисунок 2 – Результат работы программы

Таким образом мы ознакомились с теоретическими основами интерполирования, а также нами разработана программа для построения интерполяционных многочленов заданной функции, также в программе реализовано построение графика погрешности интерполяции.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Проказникова Е.Н. Конспект лекций по курсу «Вычислительная математика».
2. Авхадиев Ф.Г., Губайдуллина Р.К., Насибуллин Р.Г. Учебнометодическое пособие по численным методам анализа / Ф.Г. Авхадиев, Р.К. Губайдуллина, Р.Г. Насибуллин — Казань: Казанский (Приволжский) федеральный университет, 2019. — 113 с.
3. Федоров, Д. Ю. Программирование на языке высокого уровня Python : учеб. пособие для прикладного бакалавриата / Д. Ю. Федоров. — 2-е изд., перераб. и доп. — М. : Издательство Юрайт, 2019. — 161 с. — (Серия : Бакалавр. Прикладной курс).

УДК 004.65

И.Г. ОСИНРязанский государственный радиотехнический университет
имени В.Ф. Уткина**КРИТЕРИИ ВЫБОРА КОНФИГУРАЦИИ И
СРАВНИТЕЛЬНЫЙ АНАЛИЗ ИСКУССТВЕННЫХ
НЕЙРОННЫХ СЕТЕЙ**

Рассматриваются критерии выбора конфигурации нейронной сети для конкретной задачи, а также проводится сравнительный анализ искусственных нейронных сетей.

При выборе конфигурации ИНС для решения конкретной задачи одной из главных задач является определение количества нейронов, однако строгих соотношений, позволяющих рассчитать его, не существует.

Важным параметром, который напрямую следует из количества нейронов в сети, является количество межнейронных связей. Именно оно определяет обобщающую способность сети, а неподходящее количество межнейронных связей может привести в *недообучению* или *переобучению* сети.

Недообучением (англ. *underfitting*) ИНС называется явление, возникающее в результате обучения ИНС на недостаточно репрезентативной или слишком маленькой обучающей выборке, а также малого количества итераций обучения. Также, к аналогичному результату приведёт недостаточное количество скрытых нейронов и, как следствие, межнейронных связей в нейросети. Результатом недообучения ИНС является крайне низкая точность работы как с данными, содержащимися в выборке, так и новыми данными.

Переобучением (англ. *overfitting*) ИНС называется явление, которое возникает в результате слишком точной подстройки весов, в результате чего сеть адаптируется не к общим закономерностям в данных, а к изменениям, обусловленных случайной составляющей. Это происходит либо вследствие слишком долгого обучения, либо слишком большого количества нейронов скрытого слоя и межнейронных связей. Переобучение нейронной сети приводит к ухудшению её обобщающей способности, то есть способности работать с новыми данными, не участвовавшими в процессе обучения. Переобучение ИНС можно обнаружить, сравнивая ошибки обучения и

обобщения – если ошибка обобщения не убывает, в то время как убывает ошибка обучения, то имеет место переобучение сети.

В результате, задача подбора оптимального количества нейронов в сети сводится к поиску баланса между двумя крайностями – слишком большое количество нейронов может привести к переобучению, в то время как слишком маленькое – к неспособности сети выявлять сложные зависимости.

Существуют эмпирические рекомендации для определения конфигурации сети:

- число нейронов во входном и выходном слоях жёстко определяется числом входных и выходных переменных модели соответственно;

- число нейронов в скрытых слоях и число скрытых слоёв выбираются таким образом, чтобы количество образованных ими связей была меньше числа обучающих примеров в два-три раза.

Конфигурирование сети в соответствии с этими рекомендациями во многих случаях приведёт к получению оптимальной для решения конкретной задачи ИНС, которая не будет подвержена переобучению.

Существует множество различных алгоритмов обучения ИНС, отличающихся между собой рядом характеристик. Наиболее значимыми из них являются:

- скорость сходимости;
- вычислительная сложность;
- затраты памяти;
- простота реализации.

В этом разделе будут рассмотрены три алгоритма обучения ИНС:

- алгоритм обратного распространения ошибки (англ. *backward propagation, backprop*),

- алгоритм упругого распространения (англ. *resilient propagation, RProp*)

- алгоритм сопряжённых градиентов.

Алгоритм обратного распространения ошибки относится к градиентным алгоритмам обучения многослойного персептрона за счёт минимизации сред неквадратической ошибки выходного вектора ИНС путём итеративной коррекции весов межнейронных связей. Этот алгоритм реализует наискорейший спуск по гиперповерхности ошибок выходных значений сети. Его достоинствами является относительная

простота реализации, малые вычислительные затраты, возможность распараллеливания и относительно малые затраты памяти.

К недостаткам алгоритма обратного распространения ошибки относится присущий всем градиентным алгоритмам недостаток – они не гарантируют нахождения глобального минимума гиперповерхности ошибок, и могут застрять в локальном минимуме (рисунок 1).



Рисунок 1 – Локальный минимум на графике ошибки

Помимо этого, недостатком алгоритма обратного распространения ошибки является достаточно большая дискретность градиентного спуска, вызванная фиксированным коэффициентом обучения, которая может привести к тому, что при спуске алгоритм «перешагнёт» через локальный минимум и дальнейшие попытки спуститься будут безуспешны – алгоритм будет «шагать» меняя направление, каждый раз используя слишком большое значение коррекции, не имея возможности сильнее приблизиться к оптимуму.

Для сокращения влияния данной проблемы разработан другой градиентный алгоритм – алгоритм «упругого» распространения, представляющий собой модификацию алгоритма обратного распространения ошибки. В этом алгоритме добавлено специальное правило – если при следующем шаге спуска у частной производной поверхности ошибок меняется знак (что означает, что алгоритм «перешагнул» через минимум), это действие откатывается, а коэффициент скорости обучения уменьшается, иначе – увеличивается. Это правило позволяет избавиться от необходимости задавать глобальный параметр скорости обучения вообще. В результате, алгоритм упругого распространения ошибки показывает более быструю сходимость, однако требует больше памяти для каждого нейрона и является несколько более сложным для вычисления и реализации.

Алгоритм сопряжённых градиентов также даёт улучшение сходимости по сравнению с градиентными методами, однако он также

подвержен проблеме локального минимума. Этот алгоритм является частным случаем алгоритма сопряжённых направлений, который является популярным алгоритмом оптимизации. К его преимуществам можно отнести гарантированную сходимость к локальному минимуму за конечное число шагов, к недостаткам – сложность реализации, большие вычислительные затраты.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Галушкин А, Цыпкин Я. Нейронные сети. История развития теории. Учебное пособие. — М.: Альянс, 2015 — 840 с
2. Бакулева М.А., Корячко В.П., Орешков В.И. Интеллектуальные системы и нечёткая логика. М.: Курс. 2017.

УДК 004.318

С. В. ОСИНРязанский государственный радиотехнический университет
имени В. Ф. Уткина**МЕХАНИЗМ АУТЕНТИФИКАЦИИ УКАЗАТЕЛЕЙ В
ПРОЦЕССОРАХ НА АРХИТЕКТУРЕ ARM**

Рассматривается проблема эксплуатации уязвимостей, позволяющих перезаписать указатель в памяти, и защита от них с помощью механизма аутентификации указателей.

Повышение безопасности является одним из приоритетных направлений в развитии архитектуры микропроцессоров ARM. Одним из наиболее новых способов усложнить выполнение вредоносного кода стал механизм авторизации указателей (Pointer Authentication), описываемый спецификацией ARMv8.3-A, являющейся дополнением к ARMv8-A. Данный механизм предотвращает переход по адресу в указателе, если он был перезаписан извне, например, через уязвимость переполнения буфера.

Введение в проблему.**Пример уязвимости и её эксплуатации.**

Определённые уязвимости в ПО позволяют перезаписать часть памяти вредоносным кодом, захватив управление над программой. Примером такой уязвимости может служить переполнение буфера. Рассмотрим следующий код на C, который будет выполняться на 32-битном процессоре с архитектурой ARMv7:

```
#include <stdio.h>
#include <string.h>
void vulnerable(char *arg) {
    char buf[8];
    strcpy(buf, arg);
    printf(buf);
    printf("\n");
}
int main(int argc, char **argv) {
    vulnerable(argv[1]);
    return 0;
}
```

Эта программа при запуске считывает строку из аргумента и помещает её в массив buf, после чего выводит его содержимое на экран. При этом используется небезопасная функция strcpy(), которая

не проверяет длину записываемых данных. После компиляции ассемблерный код функции main() выглядит следующим образом:

```
(gdb) disass main
Dump of assembler code for function main:
// выполнение начинается здесь
0x000104a8 <+0>:    push   {r11, lr}
0x000104ac <+4>:    add    r11, sp, #4
0x000104b0 <+8>:    sub    sp, sp, #8
0x000104b4 <+12>:   str    r0, [r11, #-8]
0x000104b8 <+16>:   str    r1, [r11, #-12]
0x000104bc <+20>:   ldr    r3, [r11, #-12]
0x000104c0 <+24>:   add    r3, r3, #4
0x000104c4 <+28>:   ldr    r3, [r3]
0x000104c8 <+32>:   mov    r0, r3
// вызов функции
0x000104cc <+36>:   bl     0x10468 <vulnerable>
0x000104d0 <+40>:   mov    r3, #0
0x000104d4 <+44>:   mov    r0, r3
0x000104d8 <+48>:   sub    sp, r11, #4
0x000104dc <+52>:   pop    {r11, pc}
End of assembler dump.
```

При вызове функции vulnerable в стек будет записан адрес возврата, т.е. следующая инструкция (в данном случае – адрес 0x000104d0). Фрагмент стека при нормальной работе программы после вызова strcpy() выглядит следующим образом:

```
(gdb) x/8x $sp
0x7efff4e0:    0x76f9cb74    0x7efff7a6    0x41414141
0x00414141
0x7efff4f0:    0x7efff504    0x000104d0    0x7efff654
0x00000002
```

Жирным здесь выделен буфер в который была скопирована строка “AAAAAAA” (ASCII-код символа А – 0x41, и терминатор строки 0x00). Адрес возврата из функции подчёркнут.

Поскольку strcpy() не проверяет длину копируемых данных, здесь присутствует уязвимость переполнения стека. В случае ввода строки длиннее 7 знаков можно перезаписать адрес возврата. При вводе строки “AAAAAAAAAAAAABBBB” стек будет выглядеть следующим образом:

```
(gdb) x/8x $sp
0x7efff4e0:    0x76f9cb74    0x7efff79d
0x41414141    0x41414141
0x7efff4f0:    0x41414141    0x42424242
0x7efff600:    0x00000002
```

Нетрудно заметить, что адрес возврата был перезаписан буквами В. При попытке возврата из функции программа будет завершена из-за

попытки перехода по запрещённому адресу. В реальных эксплоитах такая перезапись используется для перехода к выполнению вредоносного кода, записанного далее по стеку.

Таким образом, с помощью специально сформированной строки можно подменить адрес возврата из функции и выполнить произвольный код с уровнем привилегий, соответствующим таковому у запущенной программы.

Применяемые механизмы защиты.

Приведённый выше пример является самым простым и служит исключительно для иллюстрации, т.к. современные ОС и компиляторы используют различные механизмы защиты от эксплоитов:

- Предотвращение выполнения данных (NX bit);
- Address Space Layout Randomization (ASLR);
- Программная защита стека (stack canary).

Тем не менее, несмотря на то, что данные технологии затрудняют эксплуатацию подобных уязвимостей, существуют способы их обхода. Одним из наиболее часто применяемых способов обойти NX-bit является методика Return-Oriented Programming, которая заключается в использовании легитимных фрагментов кода в качестве примитивов для выполнения произвольных действий. Также существуют различные способы обхода ASLR, зависящие от ОС.

Программная защита стека подразумевает, что помимо адреса возврата при вызове функций в стек дополнительно записывается определённое значение (canary), а при возврате из них это значение сравнивается с ожидаемым. В случае несовпадения программа прекращает работу. Такая защита увеличивает размер исполняемых файлов и негативно сказывается на производительности, поэтому многие компиляторы используют этот механизм только для части функций.

Аутентификация указателей.

Аутентификация указателей является расширением архитектуры ARMv8.3-A, доступным для 64-битных процессоров. Поскольку длина виртуальных адресов в операционных системах, как правило, значительно меньше 64 бит, в старшие разряды указателей можно записать дополнительные данные. Именно эта возможность и используется для защиты.

Pointer Authentication Code.

Коды аутентификации указателей (PAC) зависят от:

- Значения указателя;
- Значения контекста (64 бита);
- Одного из ключей (128 бит):

- APIAKey/APIBKey – для указателей на инструкции;
- APDAKey/APDBKey – для указателей на данные;
- APGAKey – для произвольных данных.

РАС рассчитывается с помощью специального криптографического алгоритма и помещается в неиспользуемые разряды указателя. В системах с использованием меток памяти (tag) это разряды от N+1 до 53, где N – длина виртуального адреса. В системах, где не используются метки, также могут быть задействованы разряды с 57 до 64.

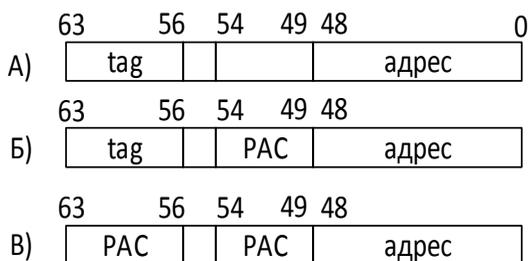


Рисунок 1 – Указатели: а) – без РАС, б) – с РАС, в) – с РАС без метки. Виртуальный адрес имеет длину 48 бит.

Применение.

Для работы с РАС были добавлены новые инструкции. Они делятся на 3 категории:

- Добавление РАС к указателю;
- Проверка РАС;
- Удаление РАС без проверки.

Кроме того, была добавлена инструкция РАСGA, которая генерирует код аутентификации из двух произвольных 64-битных значений. Эту команду можно использовать для дополнительной защиты особо важных областей памяти.

Общий алгоритм работы с РАС в случае с защитой стека выглядит следующим образом:

1. При вызове функции адрес возврата подписывается РАС с указателем на вершину стека в качестве контекста.

2. Функция сохраняет в стек адрес возврата и указатель на низ стека родительской функции, после чего выделяет место для собственных данных.

3. После завершения функции из стека загружаются ранее сохранённые адреса возврата и низа стека вызвавшей её функции.

4. Проверяется РАС адреса возврата. В случае совпадения РАС удаляется из указателя, и адрес становится действительным. В противном случае адрес возврата будет недействителен.

5. Происходит переход по адресу возврата. Если на предыдущем этапе РАС не совпал, произойдёт ошибка сегментации и программа завершит работу.

Пример использования.

В качестве примера рассмотрим простую функцию:

```
#include <stdio.h>
void fun(char* msg) {
    printf(msg);
}
```

Результат компиляции без защиты стека (используется компилятор clang 11.0.1):

```
fun(char*):
    sub    sp, sp, #32
    stp    x29, x30, [sp, #16]
    add    x29, sp, #16
    str    x0, [sp, #8]
    ldr    x0, [sp, #8]
    bl     printf
    ldp    x29, x30, [sp, #16]
    add    sp, sp, #32
    ret
```

Результат компиляции с программной защитой (-fstack-protector-all):

```
fun(char*):
    push   {r11, lr}
    mov    r11, sp
    sub    sp, sp, #16
    ldr    r1, .LCPI0_0
    str    r1, [sp, #4]
    ldr    r1, [r1]
    str    r1, [r11, #-4]
    str    r0, [sp, #8]
    ldr    r0, [sp, #8]
    bl     printf
    ldr    r0, [sp, #4]
    ldr    r0, [r0]
    ldr    r1, [r11, #-4]
    cmp    r0, r1
    bne    .LBB0_2
    b      .LBB0_1
.LBB0_1:
    mov    sp, r11
```

```

        pop    {r11, lr}
        bx     lr
.LBB0_2:
        bl     __stack_chk_fail
.LCPI0_0:
        .long  __stack_chk_guard

```

Как можно заметить, было добавлено значительное количество кода для проверки контрольного значения.

Результат компиляции с PAC (-mbranch-protection=standard):

```
fun(char*):
```

```

paciasp
sub     sp, sp, #32
stp     x29, x30, [sp, #16]
add     x29, sp, #16
str     x0, [sp, #8]
ldr     x0, [sp, #8]
bl      printf
ldp     x29, x30, [sp, #16]
add     sp, sp, #32
autiasp
ret

```

В прологе функции адрес возврата в регистре X30 (LR) подписывается с помощью функции PACIASP. Эта команда использует для подписи ключ APIAKey и регистр SP в качестве контекста. В эпилоге с помощью команды AUTIASP проверяется PAC адреса возврата. Эта команда использует те же ключ и контекст, что и PACIASP. В случае несовпадения PAC при попытке возврата из функции произойдёт ошибка сегментации и, таким образом, будет предотвращён возможный запуск вредоносного кода.

Ограничения.

В случае, если атакующий с доступом к памяти на чтение и запись может заставить систему исполнять фрагмент кода, который считывает указатель из памяти, подписывает его и сохраняет обратно (например, с помощью ROP или Jump-oriented programming), то механизм PAC будет скомпрометирован, т.к. у вредоносного кода появится возможность подписывать произвольные указатели.

Также, если у вредоносного кода есть доступ к памяти ядра и ключи не хранятся на более высоком уровне привилегий, возможно чтение ключей для подписи произвольных указателей с помощью программной реализации алгоритма генерации. Некоторые производители (к примеру, Apple) решают эту проблему применением собственного криптографического алгоритма вместо предложенного в спецификации QARMA.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Buffer Overflows: Attacks and Defenses for the Vulnerability of the Decade / C. Cowan, P. Wagle, C. Pu, S. Beattie, J. Walpole // Oregon Graduate Institute of Science & Technology. URL: https://www.cs.utexas.edu/~shmat/courses/cs380s_fall09/cowan.pdf (дата обращения: 28.03.2021)
2. Return-Oriented Programming: Systems, Languages, and Applications / R. Roemer, E. Buchanan, H. Shacham, S. Savage // University of California, San Diego. URL: <https://hovav.net/ucsd/dist/rop.pdf> (дата обращения: 26.03.2021)
3. M. Ruthland. ARMv8.3 Pointer Authentication. / Linux Security Summit, 2017. URL: https://events.static.linuxfound.org/sites/events/files/slides/slides_23.pdf (дата обращения: 23.03.2021)
4. Qualcomm Technologies, Inc. Pointer Authentication on ARMv8.3: Design and Analysis of the New Software Security Instructions. // Qualcomm Technologies, Inc., 2017 URL: <https://www.qualcomm.com/media/documents/files/whitepaper-pointer-authentication-on-armv8-3.pdf> (дата обращения: 23.03.2021)
5. B. Azad. Examining Pointer Authentication on the iPhone XS. // Google Project Zero, 2019. URL: <https://googleprojectzero.blogspot.com/2019/02/examining-pointer-authentication-on.html> (дата обращения: 24.03.2021)
6. Arm Limited .Arm® Architecture Reference Manual: Armv8, for Armv8-A architecture profile. 2017

УДК 623.4.018

Д.С. ПАНИНСКАЯ, В.В. СУСКИН

Рязанский государственный радиотехнический университет
имени В. Ф. Уткина

УСТРОЙСТВО СТЫКОВКИ И ПЕРЕМЕЩЕНИЯ ЧЕКИ СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ

Рассматривается принцип работы устройства стыковки и перемещения чеки специального назначения.

Устройство стыковки и перемещения чеки специального назначения предназначено для диагностики изделий (боевых ракет). Изделие имеет в своей конструкции приспособление чеки – нажимное устройство и электромагнит ЭМТ-90А2. Исходным положением чеки является нижнее положение (чека нажата) – ракета находится в контейнере, в котором изначально и располагается. Противоположное положение – верхнее (чека отжата) – ракета находится в воздухе.

Из-за неудобства диагностики изделия непосредственно в контейнере (большие габариты, подстыковка множества систем диагностики) было разработано устройство стыковки и перемещения чеки специального назначения. Данное устройство помещается на изделие, фиксируется, далее подключаются системы диагностирования, которые объединяются и управляются автоматизированной системой контроля (АСК). Поднятие чеки изделия (ракета находится в воздухе) является пуском для систем диагностики, начинается проверка различных параметров (стабилизация полета, управление расходом топлива и другие).

Устройство стыковки и перемещения чеки специального назначения реализуется на поступательном электромеханизме МП4С10А04. Шток устройства, связанный с электромеханизмом, совершает поступательное движение.

Далее произведены расчеты времени с момента подачи команды до момента установки чеки в нижнее положение и времени с момента снятия команды до момента возврата чеки в исходное состояние, они, согласно техническому заданию, не должны превышать 3,8 секунды, и 0,2 секунды соответственно.

Перемещение чеки изделия в нижнее положение будет иметь вид:

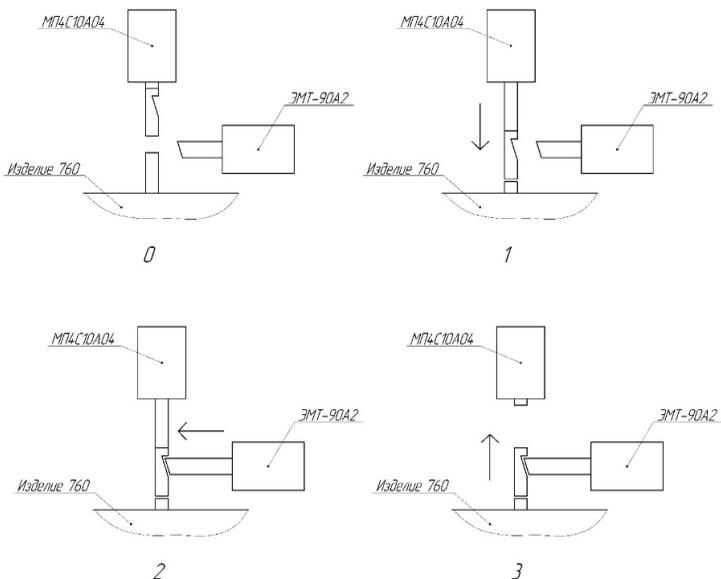


Рисунок 1 – Перемещение чеки в нижнее положение:

0 - исходное положение;

1 – опускание штока изделия электромеханизмом;

2 – зажатие штока изделия в нижнем положении электромагнитом;

3 – возвращение штока электромеханизма в исходное положение.

Нажатие (худший вариант: скорость электромеханизма 8мм/с)

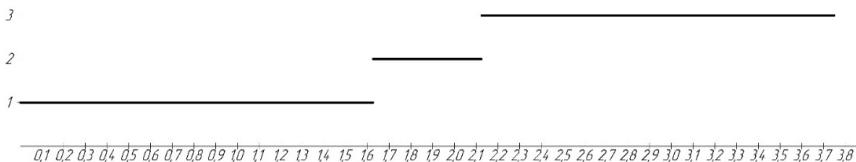


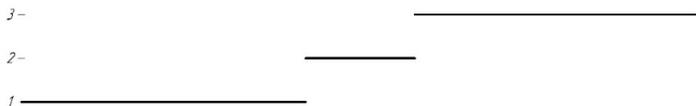
Рисунок 2 – Вариант нажатия в худшем случае:

1 – опускание штока изделия электромеханизмом;

2 – зажатие штока изделия в нижнем положении электромагнитом;

3 – возвращение штока электромеханизма в исходное положение.

Нажатие (номинальный вариант: скорость электромеханизма номинальная 10 мм/с) в секундах:

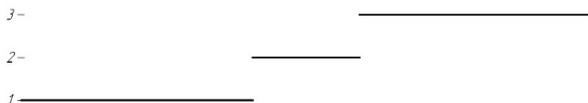


01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38

Рисунок 3 – Вариант нажатия номинальный:

- 1 – опускание штока изделия электромеханизмом;
- 2 – зажатие штока изделия в нижнем положении электромагнитом;
- 3 – возвращение штока электромеханизма в исходное положение.

Нажатие (лучший вариант: скорость электромеханизма 12 мм/с) в секундах:



01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38

Рисунок 4 – Лучший вариант нажатия:

- 1 – опускание штока изделия электромеханизмом;
- 2 – зажатие штока изделия в нижнем положении электромагнитом;
- 3 – возвращение штока электромеханизма в исходное положение.

Отжатие пружины в секундах будет иметь вид:

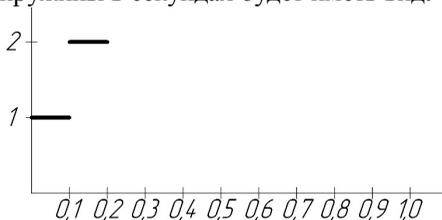


Рисунок 5 – Отжатие пружины:

- 1 – отпуская штока изделия в нижнем положении электромагнитом ЭМТ-90А2-А;
- 2 – отжатие пружины.

Перемещение чеки изделия в верхнее положение будет иметь вид:

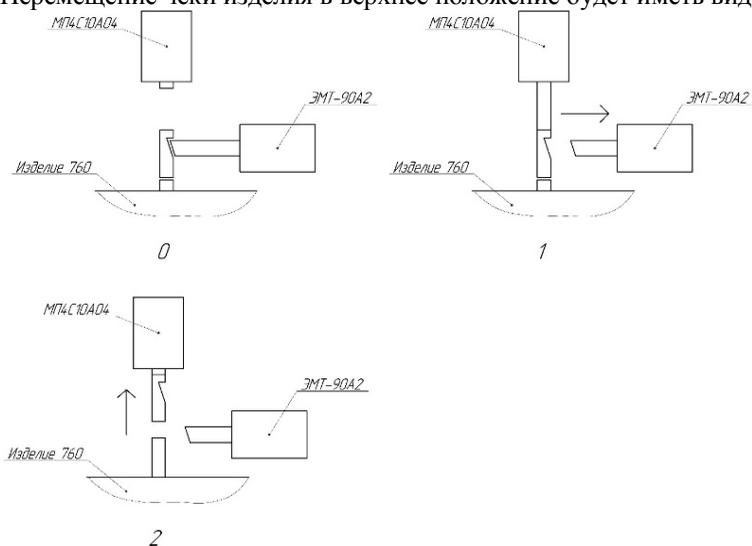


Рисунок 6 – Перемещение чеки в верхнее положение:

- 0 - исходное положение;
- 1 – отпускание штока изделия в нижнем положении электромагнитом ЭМТ-90А2-А;
- 2 – отжатие пружины.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Гольдштейн Б. С. Системы коммутации. СПб.: ВHV-Санкт-Петербург. - 2003.

УДК.621.311.24

Г.В. ПОВЕРЕННЫЙ

Рязанский государственный радиотехнический университет
имени В. Ф. Уткина

**ИСПОЛЬЗОВАНИЕ ВЕТРОУСТАНОВКИ В СИСТЕМЕ
ЭЛЕКТРОСНАБЖЕНИЯ**

В статье предложены оригинальные схемные решения для эффективного использования ветро-энергетических установок при их совместной работе с системами централизованного электроснабжения, позволяющие максимально использовать энергию ветра для выработки качественной электрической энергии, экономить её потребление от централизованного источника и снизить потери напряжения и мощности в линиях электропередач.

Все выпускаемые ветроустановки отечественного и иностранного производства рассчитаны на выработку качественной электроэнергии номинальной мощности ВЭУ при расчётной скорости ветра, колеблющуюся в пределах от 9 до 13 м/с, в зависимости от типа ВЭУ. Выработка некачественной электроэнергии генератором ВЭУ начинается при минимальной скорости ветра, равной 2–3 м/с. Поэтому основным показателем эффективного использования ВЭУ является выбор района с высоким показателем среднегодовой скорости ветра.

В качестве примера рассмотрим ветровой ресурс Челябинской области. По статистическим данным скорости ветра, область делится на четыре ветровых района. Первый район включает в себя открытые вершины Уральских гор, которые расположены на северо-западе области. Среднегодовая скорость ветра в этом районе составляет 10–13 м/с. Второй ветровой район расположен на юге, юго-западе региона и охватывает более 50% территории области со средней скоростью ветра 4,5 м/с. Данная зона имеет развитое сельское хозяйство. Третий район охватывает центральную и северную часть региона и занимает примерно 30% области. Этот район расположен в лесостепной зоне со средней скоростью ветра 2,5–3 м/с. Четвёртый район расположен в горнозаводской зоне, где средняя скорость ветра составляет 1,5–2 м/с.

Из сравнения среднегодовых скоростей ветра и расчётных скоростей ВЭУ видно, что первый ветровой район эффективен для применения существующих ВЭУ. Применение ВЭУ во второй и третьей ветровых зонах является менее эффективным, так как низкое поступление ветрового потока приводит к выработке некачественной электроэнергии. Без применения специальных устройств, предназначенных для улучшения качества электроэнергии,

вырабатываемую энергию можно использовать только для теплоснабжения. Использование ВЭУ в четвёртом районе возможно неэффективно, так как ожидается минимальная выработка электрической энергии.

Для увеличения производства качественной электроэнергии во втором и третьем регионах разработаны схемные решения, позволяющие преобразовывать электроэнергию, вырабатываемую ветрогенератором при скорости ветра ниже расчетной. Принцип действия этих цепей основан на преобразовании генерируемого переменного напряжения переменной частоты в зависимости от скорости ветра в выпрямленное пульсирующее напряжение, которое впоследствии инвертируется в переменное напряжение постоянной частоты 50 Гц. Для обеспечения постоянного значения напряжения на выходе установки используются преобразователи импульсов постоянного напряжения, стабилизаторы и управляемые выпрямители.

Для реализации этого принципа преобразования энергии были разработаны различные схемные решения. Однако при использовании управляемых выпрямителей невозможно максимизировать энергию, вырабатываемую генератором. Использование стабилизатора напряжения за генератором влечет за собой большие потери мощности на частоте ниже номинальной [3]. В известных схемах выходные параметры напряжения и тока низкие или не соответствуют нормам качества электрической энергии по синусоидальности из-за инверсии выпрямленного пульсирующего напряжения. Использование сглаживающего фильтра для уменьшения пульсаций не позволяет добиться желаемого результата из-за несоответствия частоты и величины выпрямленного напряжения [2].

На рисунке 1 представлена схема использования ВЭУ в системе электроснабжения.

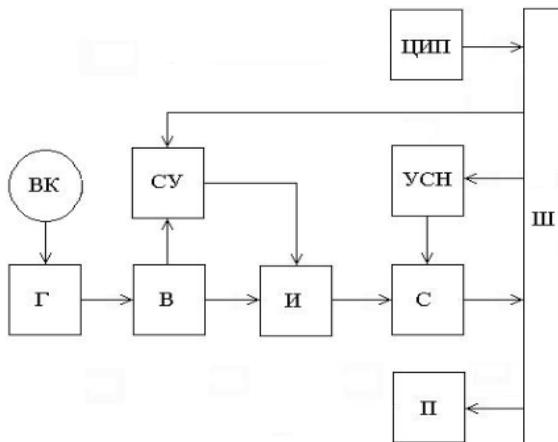


Рисунок 1 – Схема использования ВЭУ в системе электроснабжения

Схема использования ВЭУ в системе электроснабжения включает следующие обозначения: ВК –ветроколесо; Г –генератор; СУ –сглаживающее устройство; УСН –устройство согласования величин напряжений ВЭУ и централизованного источника питания; В – неуправляемый выпрямитель; И –инвертор; С –стабилизатор; Ш – общие шины; ЦИП –центральный источник питания; П –потребитель; ЦИП –централизованный источник питания; П –потребитель.

Перечисленные недостатки не позволяют подключить ВЭУ напрямую к электрической сети, через которую осуществляется централизованное электроснабжение потребителей. В результате ветроустановки, выпускаемые в основном, работают вместе с централизованным источником электроснабжения по отдельной схеме или используются для автономного электроснабжения потребителей вместе с аккумуляторными батареями. Для устранения перечисленных недостатков предлагается схема (рис.), позволяющая максимально использовать энергию ветрового потока и обеспечивать потребителей качественной электрической энергией [1].

Предлагаемая ветряная турбина состоит из ветряного колеса и синхронного генератора, подключенного к инвертору, который дополнительно содержит сглаживающее устройство и устройство согласования напряжений централизованного источника энергии и ветряной турбины. Инвертор основан на силовых транзисторах, которые представляют собой полностью управляемые полупроводниковые устройства. За счет подачи выпрямленного

пульсирующего напряжения на вход инвертора сглаживающее устройство, подключенное к общим шинам, входным и управляющим цепям инвертора, корректирует управляющий сигнал инвертора в зависимости от величины и коэффициента пульсаций выпрямленное напряжение, позволяющее получить на выходе инвертора синусоидальное напряжение. Это дает возможность подключать инвертор к общим шинам через стабилизатор напряжения, что позволяет поддерживать номинальное напряжение на выходе установки. К стабилизатору напряжения и общим шинам подключено устройство согласования напряжения централизованного источника и напряжения на выходе установки, которое корректирует значение выходного напряжения установки в зависимости от напряжения на общих шинах. Для предотвращения подачи в установку обратного тока от сети при несоответствии значений генерируемого установкой напряжения напряжению на шинах между общими шинами и напряжением устанавливается реле обратного тока. стабилизатор.

В случае сбоя питания централизованной системы обслуживающим персоналом или с помощью релейной защиты сглаживающее устройство теряет питание от шин, в результате чего выработка электрической энергии немедленно прекращается, что не позволяет установке работать в течение длительного времени. короткое замыкание.

Таким образом, предложенная схема использования ветряных турбин позволяет максимально использовать энергию ветра для выработки качественной электрической энергии, сэкономить ее потребление из централизованного источника и снизить потери напряжения и мощности в ЛЭП.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Пат. 2325551 Российская Федерация. Устройство для автономного энергоснабжения потребителей / С.К. Шерьязов, А.А. Аверин. Опубл. 26.12.2006 г.
2. Пат. 2221165 Российская Федерация. Ветроэлектрическая станция / Лейён Матс, Кюландер Гуннар. Опубл. 01.10.2004 г.
3. Пат. 89184 Российская Федерация. Ветроэлектрическая установка / С.К. Шерьязов, М.В. Шелубаев. Опубл. 27.11.2009 г.

УДК.621.311.24

Г.В. ПОВЕРЕННЫЙРязанский государственный радиотехнический университет
имени В. Ф. Уткина**ЦЕЛЕСООБРАЗНОСТЬ ВНЕДРЕНИЯ
ВЕТРОЭНЕРГЕТИЧЕСКИХ УСТАНОВОК
В УСЛОВИЯХ РОССИИ***Анализ схем автономного электроснабжения на базе ВЭУ и анализ целесообразности внедрения ветроэнергетических установок в условиях России.*

Любая техническая и технологическая инновация должна сопровождаться экономической оценкой целесообразности ее использования, поскольку именно рентабельность для потребителя определяет возможность внедрения того или иного продукта в рыночных условиях.

Цена на электроэнергию, полученную традиционными методами, неуклонно растет, в связи с чем становится все более актуальным вопрос получения ее альтернативными методами, среди которых, как отмечают многие специалисты, наиболее перспективным является использование ветроэнергетических установок (ВЭС). и оптимально. Как известно, Россия - страна с самой протяженной морской береговой линией, большой площадью пресноводных озер, обширными равнинными территориями, что является несомненным потенциалом для развития технологий производства электроэнергии на основе использования кинетической энергии ветра. Практический опыт стран с близкими к российским геоклиматическими условиями, таких как Канада, Германия, США, Дания и др., доказывает возможность этого в коммерческих целях.

Ветряные электростанции принято делить на две большие группы: сетевые и автономные (более мелкие). В сеть входят ветряные турбины, предназначенные для работы параллельно с единой энергосистемой. По данным зарубежных источников, сетевые ветряки окупаются за 6-8 лет, а потом приносят чистую прибыль. В настоящее время в России действует только одна ветровая электростанция сетевого типа - ВЭС Воркута, входящая в энергосистему Комиэнерго. Однако он практически не используется и заброшен. Отсутствие спроса на ветряные турбины в основном связано с все еще достаточно низкими ценами на электроэнергию в России, отсутствием

соответствующего законодательства и стандартов, а также монополией магистральных сетей.

Это создает предпосылки для изучения вопроса о возможности внедрения автономных электростанций, мощность которых соизмерима с мощностью потребителей и которые схематично не связаны с «большой» электроэнергетикой. Стоимость 1 кВтч электроэнергии, получаемой автономными установками, в настоящее время довольно высока по сравнению с дизельными электростанциями, что затрудняет их внедрение, поэтому этому вопросу необходимо уделить особое внимание, чтобы реализовать их резерв эффективности, чтобы минимизировать затраты на генерацию. электричество. Решение этого вопроса позволит широко внедрить эти технологии в различных отраслях народного хозяйства.

При строительстве автоматизированной электростанции на базе ветряной турбины важнейшее значение имеет комплексное изучение всех вопросов, поскольку ветровые турбины должны снижать:

- финансовые затраты на транспортировку и хранение топлива;
- потери при передаче электрической и тепловой энергии от источника к потребителю;
- затраты на эксплуатацию и техническое обслуживание энергоисточников.

Наряду с достоинствами ВЭУ существует и ряд их недостатков. Рассмотрим основные причины, сдерживающие повсеместное внедрение ВЭУ. К объективным причинам медленного внедрения ВЭУ в практическую энергетику относятся:

- ветер как источник энергии. Ветер обладает непостоянными характеристиками, имеет большие колебания скорости, сопровождающиеся резким изменением выдаваемой мощности, средние скорости ветра существенно изменяются в суточном и годовом циклах. Мировая практика показывает, что при среднегодовых скоростях ветра менее 4–5 м/с применение сетевых ВЭУ неэффективно. Согласно ветровому кадастру России, лишь 40% её территории может использоваться для выработки электроэнергии. Значительным ветроэнергетическим потенциалом обладают зоны побережья и островов Северного Ледовитого и Тихого океанов, Азово-Черноморская и Каспийская зоны;

- особенности преобразования энергии ветра в электрическую. В результате непостоянства ветра и сильной зависимости мощности от скорости ветра ВЭУ не могут обеспечить высокое качество электроэнергии и надёжность электроснабжения потребителей в автономном режиме. Число часов использования генерирующей

мощности ВЭУ зависит от среднегодовой скорости ветра и лежит в пределах 2–4 тыс. ч в год. Наиболее благоприятны для работы ВЭУ осенне-зимний и ранний весенний периоды года, что в целом совпадает с условиями изменения электрической и тепловой нагрузок объектов автономного энергоснабжения (ОАЭ);

- высокая стоимость. По данным различных источников, стоимость 1 кВт вводимой в эксплуатацию мощности ВЭУ составляет от 1000 до 1500 долл. США, что в несколько раз превышает капиталовложения в дизельные электростанции небольшой мощности (до 300 кВт), составляющие 200–250 долл/кВт. По оценкам экспертов, в перспективе по мере развития ветротехники можно ожидать снижения стоимости ВЭУ.

Проведём анализ основных схемы с использованием коммутатора и штатного инвертора.

Схема с использованием коммутатора и штатного инвертора (рисунок 1).

Потребитель получает электроэнергию из наиболее выгодного источника. Например, при наличии ветра и нормальной работе ВЭУ потребитель питается от системы «Ветроустановка + аккумулятор», при отсутствии ветра - от сети, бензинового (дизельного) генератора или других источников питания (солнечные батареи и др.), в соответствии с очередностью входа в выключатель.

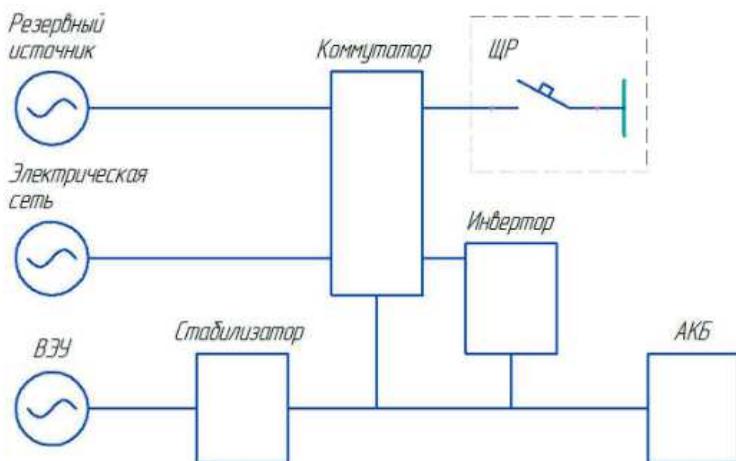


Рисунок. 1 – Структурная схема с использованием коммутатора и штатного инвертора: АКБ – аккумуляторные батареи

Преимущества:

- переключение происходит автоматически;
- недорогая система, не требуется дополнительных приборов.

Недостатки:

- при переключении входов коммутатора из одного положения в другое происходит несущественный перебой в энергоснабжении (до 20 миллисекунд);

- максимальная потребляемая мощность при питании от ВЭУ не может быть больше номинальной мощности инвертора.

Таким образом, в результате анализа схем автономного электроснабжения на базе ВЭУ нами был сделан следующий вывод, что наиболее целесообразной к внедрению с технико-экономической точки зрения является схема с использованием коммутатора и штатного инвертора ввиду наличия обратных связей и малых потерь электрической энергии.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Безруких П.П. Что может дать энергия ветра. М., 2002. 36 с.
2. Фатеев Е.М. Ветро двигатели и ветроустановки. М.: ОГИЗ, 1948. 539 с.
3. Сибикин Ю.Д., Сибикин М.Ю. Нетрадиционные и возобновляемые источники энергии: учебное пособие. М.: КНОРУС, 2010. 232 с.
4. Закржевский Э.Р. Ветро двигатели для механизации животноводческих ферм. Минск: БССР, 1959. 197 с.
5. Модификация 6-лопастной ВЭУ – малая ветроустановка для обеспечения энергопитания небольшого дома, удалённого объекта.
URL: <http://do.gendocs.ru/docs/index-365684.html?page=2>

УДК 004.772

Н.С. ПОТАПКИНА

Рязанский государственный радиотехнический университет
имени В. Ф. Уткина

**КОНСТРУИРОВАНИЕ И ПРОЕКТИРОВАНИЕ
ВЫЧИСЛИТЕЛЬНОГО КОМПЛЕКСА
ПЕРЕДАЧИ И ОБРАБОТКИ ДАННЫХ В
ПРОГРАММНО-КОНФИГУРИРУЕМЫХ СЕТЯХ**

Рассматривается конструирование и проектирование вычислительного комплекса на основе плат Arduino Uno и Ethernet Shield, основные достоинства и области применения данного комплекса. Также рассматривается возможность и путь создания «умного» университета.

Программно-конфигурируемые сети (ПКС) – это архитектура, развивающаяся и подстраиваемая под постоянно меняющиеся современные системы и приложения [1].

ПКС отделяет функции передачи данных от функций управления в дискретных сетевых устройствах. Данный вид сетей дает возможность отдельно программировать управление сетью и структуру приложений. Главное отличие ПКС от обычной сетевой архитектуры заключается в управлении коммутатором организацией сетевых функций [1]. Данный вид управления происходит в любом месте, независимо от конкретных сетевых соединений между сервером и устройствами. В то же время отдельные устройства не принимают решения о трафике.

Использование ПКС дает возможность разработать, например, систему контроля за состоянием технологического процесса (контроль температуры, влажности, чистоты, количества серы в процессе переработки нефтепродуктов) и собирать данные с датчиков состояния на одном сервере, с помощью которого можно контролировать весь технологических процесс производства. Также ПКС применяются при разработках умного дома, чтобы в любой момент из любой точки с доступом в сеть можно было не только выявлять включен или выключен утюг или свет, но и управлять этим.

В данном случае рассматривается устройство, с возможностью установки во все аудитории университета для отслеживания влажности и температуры в помещениях, контроля открытия/закрытия дверей и окон, наличия открытого огня и сигнализации об этом. Все эти данные будут собираться на одном сервере с возможностью отслеживания в режиме «онлайн» как работниками университета, так и студентами, и родителями.

Для контроля открытия/закрытия дверей и окон потребуются герконовые датчики (герконы). Принцип работы данных датчиков основан на изменении состояния двух разомкнутых или замкнутых контактов в колбе с инертным газом или с вакуумом под воздействием магнитного поля (магнита).

В зависимости от типа герконы могут быть замыкающими (под воздействием магнитного поля контакты в колбе замыкаются), размыкающими (под воздействием магнитного поля контакты в колбе размыкаются) и переключающими (по принципу работы ключа). Соответственно, устанавливая датчик и магнит на неподвижную и открывающуюся части дверей и окон, можно отслеживать наличие магнитного поля (в данном случае оно будет сигнализировать об открытии/закрытии). Для приема сигнала с датчика используется цифровой порт.

Помимо подстроечного резистора, используемого для настройки чувствительности, цифровой датчик геркона для Ардуино содержит компаратор (микросхему) LM393. С помощью компаратора происходит генерирование цифрового сигнала при срабатывании геркона. Соответственно, при наличии нуля, будет выдаваться сообщение «дверь/окно закрыто», при наличии единицы – открыто.

Датчик огня (flame sensor Arduino) позволяет определить наличие открытого пламени с помощью инфракрасного приемника. На основе датчика создается пожарная сигнализация в аудиториях.

На датчике загорается индикатор, при наличии нуля выдается сообщение «пожар», если обнаружено пламя на расстоянии до 1 метра от сенсора, и «отсутствует открытый огонь» - при отсутствии пламени в пределах видимости инфракрасного приемника.

Аналоговый выход, имеющийся на датчике, информирует о наличии сигнала и его характеристике. Таким образом, с помощью сенсора можно определять не только наличие открытого огня, но и то, настолько он сильный, а также площадь его распространения. ИК датчик реагирует излучение в диапазоне 750 — 1100 нм [2].

Для отслеживания влажности и температуры в аудиториях требуется датчик DHT22. Для измерения температуры имеется термистор, для измерения влажности воздуха – емкостной датчик влажности. Данный датчик может выполнять аналого-цифровые преобразования и выдавать цифровой сигнал, который считывается с помощью микроконтроллера.

Термистор представляет собой специальный резистор, изготовленный на основе смешанных оксидов переходных металлов, сопротивление которого изменяется с изменением температуры, т.е.

увеличение температуры приводит к падению сопротивления на датчике [3].

Емкостной датчик влажности представляет собой конденсатор с переменной емкостью. Конденсатор датчика находится в герметичном чехле с влагопоглощающим слоем и содержит токопроводящие обкладки из медной фольги на текстолите. При попадании частиц воды на этот слой, меняется его диэлектрическая проницаемость, и, соответственно, изменяется емкость конденсатора.

Датчик DHT22 имеет напряжение от 3 до 5 Вольт, максимальный ток при преобразовании – 2,5 мА, интервал измерения влажности - 0% - 100%, интервал измерения температуры от -40 –до +125 градусов по Цельсию, период измерения - 2 секунды, частота – до 0,5 ГЦ.

После загрузки скетча и подключения датчика, результат измерений можно посмотреть в окне монитора порта. Там будут выводиться значения температуры и влажности

Все полученные результаты с датчиков можно посмотреть в окне монитора порта, сохранить и вывести на ЖК-экран, послать через bluetooth или интернет.

Для передачи собранных с датчиков данных по сети требуется еще одно устройство – Ethernet Shield. Он позволяет легко подключить плату к локальной сети или сети Интернет, предоставляет возможность ардуино отправлять и принимать данные из любой точки мира через интернет-соединение. После подсоединения контактов платы Ethernet Shield к ардуино, остается только подключить шилд через витую пару RJ45 разъемом, например, к Wi-fi роутеру. Выполнив физическое подключение остается написать и загрузить в плату соответствующий скетч, который превратит плату в сервер, отвечающий на запросы удаленных устройств или веб-клиента, собирающего данные с серверов или отправляющего на них эти данные.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Коннов А.Л., Анализ и проектирование программно-конфигурируемых сетей: учебное пособие / А.Л. Коннов – Оренбург: ОГУ, 2016. – 115 с.
2. Максимов Н.В., Компьютерные сети / Н.В. Максимов – Москва: Форум, 2013. – 464 с.
3. Кузьменко Н.Г., Компьютерные сети и сетевые технологии / Н.Г. Кузьменко – Санкт-Петербург: Наука и Техника, 2013. – 368 с.

УДК 004.415.53

Р.А. ПРИХОДЬКО

Севастопольский государственный университет

**О ПЕРЕХОДЕ К МИКРОСЕРВИСНОЙ АРХИТЕКТУРЕ
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

*Проведено сравнение монолитной и микросервисной архитектуры приложений.
Проведен анализ метода перехода от монолитной архитектуры к микросервисной.*

Известно [1], что программное обеспечение (ПО) можно считать имеющим монолитную архитектуру (МА), если оно разрабатывается, тестируется и разворачивается как единое целое. ПО считается имеющим микросервисную архитектуру (МСА), если состоит из специализированных приложений, разрабатываемых, тестируемых и разворачиваемых независимо друг от друга. Никакая из этих архитектур не является безусловно лучше другой – каждая имеет свои преимущества и недостатки.

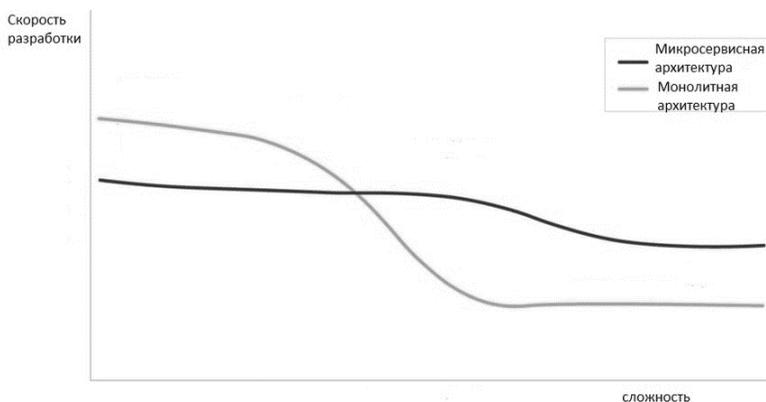


Рисунок 1 – Сравнение скорости разработки в монолитной и микросервисной архитектуре

На ранних этапах развития ПО, когда кодовая база не велика и функционал ограничен, МА показывает более быстрые по сравнению с МСА темпы разработки (рисунок 1). Однако, по прошествии времени, получая разветвленный функционал и большую кодовую базу, такое ПО начинает иметь следующие недостатки:

1. Запутанность кода
2. Сложная процедура выпуска программного продукта
3. Сложность внедрения новичка на проект
4. Необходимость обновлять ПО только целиком даже при небольших изменениях
5. Плохая масштабируемость
6. Невозможность внедрения новых фреймворков, сложность обновления библиотек

Решением перечисленных проблем является переход к МСА [2]. Таким образом, МСА можно считать эволюционным развитием монолитной. Создание нового ПО на месте старого реализовано быть не может - нельзя переходить от одной архитектуры к другой сразу, т. к. такой подход дорогостоящ и имеет неопределенные перспективы. Переход должен осуществляться постепенно. Актуальным является вопрос выбора методики перехода от МА к МСА. Существует два основных метода такого перехода [3]:

1. Реализовывать новый функционал в виде отдельных микросервисов (МС)
2. Постепенно выносить существующий функционал в МС, уменьшая монолитное приложение до тех пор, пока оно не превратится в МС или не исчезнет совсем

Второй подход называется «шаблоном душителя» и является основным при переходе к МСА. Поскольку ядром любого приложения является доменная модель (ДМ), перед реализацией выделения функционала в отдельные МС следует вначале выделить в ДМ логически законченные подмодели - так называемые ограниченные контексты (ОК) [4]. Ограниченный контекст определяет область применения модели, дает понять для каких целей используется модель, каким задачам должна соответствовать и какими данными может пренебречь. Он также гарантирует, что предметные понятия, выходящие за пределы данного контекста модели, не будут отвлекать от задач, для решения которых она предназначена и явно сообщает разработчикам, за что отвечает модель, а за что не отвечает. В качестве ОК отделяют вначале служебный, а потому логически независимый функционал, такой, как авторизация, обработка сохраняемых файлов, работа с уведомлениями и т. д. Для выделения в ОК частей основного функционала проводят серьезный анализ.

После выделения ОК последовательно делаются следующие шаги:

1. Модель данного ОК изолируется от основной модели

2. Часть базы данных (БД), соответствующая выделяемой модели, отделяется от основной БД.

3. Логика, работающая с моделью ОК, изолируется от основной

При изоляции модели объектные ссылки на сущности другого контекста заменяются идентификаторами. Идентификаторы служат интерфейсными частями ОК: модели различных сервисов могут иметь информацию только об идентификаторах моделей других сервисов, но не больше.

БД, соответствующая выделяемой модели, отделяется от основной БД монолита следующим образом:

1. Если таблица содержит данные разных ОК, ее разбивают на несколько различных таблиц (рефакторинг «Разбиение таблицы» [5])

2. Если таблицы различных ОК связаны между собой внешними ключами, ключи удаляют, оставляя только идентификаторы

Если структура выделяемой части БД удовлетворяет потребностям текущего дня, ее оставляют как есть. Если же структура признается устаревшей, ее меняют и в новый МС попадает уже новая структура.

Логика, оперирующая с выделяемым ОК, чаще всего тесно связана с общей логикой и имеет с ней общие части. При выделении логики следует придерживаться следующих этапов:

1. Если некий класс содержит логику, работающую с различными ОК, его разделяют на различные классы.

2. Если некий метод содержит логику, работающую с различными ОК, его разделяют на различные методы.

3. Если логика будущего МС вызывается из логики другого ОК, вызов перенаправляют через внешний АПИ.

4. Общие части логики различных ОК выносят внешними ссылками (например, в виде `inet` – пакетов)

Как модель с кодом, так и БД будущего МС, будучи изолированными, могут по-прежнему располагаться в основном монолитном приложении. Их вынесение в отдельное место является отдельным этапом, который осуществляется по-разному. Так, сначала может быть вынесена БД, а модель с кодом какое-то время могут работать, будучи не вынесенными. Можно вынести сразу все компоненты, удалив из в старом местоположении. Также практикуется вынесение без удаления – в этом случае используется и новая часть в составе МС, и старая в рамках монолитного приложения. Последний вариант хорош тогда, когда нужно осуществить переход на новую архитектуру постепенно, например вначале перевести на нее новых пользователей или только некоторые виды процессов. Постепенный

переход безопасен, оставление «обратного хода» позволяет в случае непредвиденных обстоятельств переключиться на старый функционал, для того чтобы доработать внезапно возникшие проблемы в новом. Реализуется переключение в виде параметров конфигурационных файлов.

С точки зрения хранения кода МС существует два подхода:

1. Код всех МС хранится в одном репозитории
2. Коды всех МС разнесены в разные репозитории

Первый подход хорош тем, что все МС могут использовать один и тот же общий код, хранящийся в этом же репозитории. Однако МС с неразделенным кодом неудобно выпускать: ветвь с уже законченным функционалом одного МС может содержать незаконченный функционал другого. Также неразделенный код различных МС рано или поздно перемешивается, и архитектура начинает приближаться к МА. Второй подход исключает последнее, но использование общего кода становится проблемой. Здесь решением является или дублирование кода, или вынесение в подключаемые пакеты `nuget`, что снижает скорость разработки. Решение о способе хранения кода следует выбирать индивидуально, взвесив все «за» и «против».

Важным при выделении МС является проектирование взаимодействий между МС и остальным приложением [6].

Выделяют два основных вида таких взаимодействий:

1. Общая БД
2. Общие файлы
3. Синхронные вызовы
4. Обмен сообщениями

Общая БД дает слабую связность контекстов. Однако имеет недостатки:

1. БД является слабым звеном сразу двух сервисов
2. Оба сервиса конкурируют за ресурсы БД
3. Возникают проблемы обработки ошибок БД

Интеграция через файлы также дает слабую связность контекстов и не имеет проблем с блокировкой, однако является «самодельным» решением, влекущим проблемы разработки своих форматов взаимодействия.

Первые два типа считаются временными.

Синхронные вызовы используются в случаях, когда на запрос нужен немедленный ответ (RPC, REST). Они простоты в применении и универсальны. Однако не обеспечивают надежность передачи сообщений и могут привести появлению «цепочек МС» - многократной последовательности синхронных вызовов от МС к МС,

вызывающей таймауты обработки запроса в силу его долгой обработки.

Асинхронное взаимодействие используется в случае, когда немедленный ответ не требуется. В качестве транспорта используется брокер сообщений (RabbitMQ, Kafka и др.). Событие передается как объект, который кладется в очередь брокера сообщений и ждет своей обработки. Преимущество заключается в обеспечении надежной передачи сообщений. Недостатки – в громоздкой реализации и в практической невозможности реализовать синхронное взаимодействие.

Виды взаимодействия выбираются в зависимости от специфики запросов на основе тщательного анализа.

Рассмотрены недостатки приложения с МА, обозначена актуальность проблемы правильного перехода от МА к МСА. Рассмотрен метод решения данной проблемы.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Ричардсон К. Микросервисы. Паттерны разработки и рефакторинга. СПб.: Питер, 2019. 544 с
2. Приходько Р.А. О методе формирования микросервисной архитектуры бизнес-приложений// Информационные технологии. 2019. С. 201 – 206.
3. Миллетт С., Тьюн Н. Предметно-ориентированное проектирование. Паттерны, принципы и методы. СПб.: Питер, 2017. 832 с
4. Эванс Э. Предметно-ориентированное проектирование. Структуризация сложных программных систем. М.: Вильямс, 2018. 448 с
5. Эмблер С. Рефакторинг баз данных. М.: Вильямс, 2016. 368 с
6. Ньюман С. Создание микросервисов. СПб.: Питер, 2016. 304 с

УДК 004.023

И.Д. РАЗМАХНИНРязанский государственный радиотехнический университет
имени В.Ф. Уткина**РАЗРАБОТКА ПРОГРАММНЫХ СРЕДСТВ КОМПОНОВКИ
ЭЛЕМЕНТОВ С ИСПОЛЬЗОВАНИЕМ МАТРИЧНОГО
АЛГОРИТМА**

В данной статье будет рассмотрены актуальность, значимость и важность программных средств компоновки элементов с использованием матричного алгоритма в современных системах проектирования.

Современные системы проектирования и разработки печатных плат не являются полностью автоматизированными. Непосредственное участие человека в процессе проектирования, и в последующем изготовлении продукта по сей день является обязательным. Задача компоновки является одной из основных задач в проектировании печатных плат, ведь она исключает паразитные электрические взаимосвязи между элементами, тем самым повышая качество финального продукта.

Одной из самых перспективных областей использования программных продуктов для компоновки элементов являются автоматизация управления и проектирования. Использование таких продуктов позволяет значительно сократить время разработки продукта, повысить качество разрабатываемого продукта, ну и конечно же фактор снижения затрат на производство тоже играет огромную роль в данном вопросе.

Стоит отметить важную проблему при создании современных систем автоматизированного проектирования ресурсов в виде необходимости перевода так называемого «творческого процесса» в четкую структурированную систему, работа которой напрямую зависит от строго обоснованных математических решений.

Использование таких алгоритмов в создании САПР является залогом создания качественной системы, которая будет действительно улучшать процесс создания и производства оборудования.

Математические модели являются основой создания данных методов, а также широко используются алгоритмы преобразования данных моделей для дальнейшего повышения и улучшения качества продукта.

Для большей части объектов проектирования в настоящее время используются математические модели графов, а для их оптимизации, используются эвристические методы.

Этот процесс приводит к локальному оптимуму и не оптимальным результатам проектирования. Данная ситуация обычно возникает при произвольном соединении по числу элементов группы.

Рассмотрим суть задачи компоновки. Она заключается в формировании отдельных групп из элементов дискретной системы, возникая при отборе элементов электрической схемы, помещаемых в отдельную конструктивную единицу, при сборке конструктивных элементов низшего в конструктивные элементы высшего уровней.

Одним из наиболее важных критериев задачи компоновки является оптимизация суммарного числа связей, проходящих между отдельно сформированными группами элементов. Под оптимизацией данного критерия обычно подразумевается минимизация, но следует также отметить, что существуют такие задачи, где требуется получить максимум определяемых связей. Стремление уменьшить количество связей можно трактовать, как увеличение «плотности» связей внутри каждой группы.

Также следует отметить возможные ограничения при выполнении решения данной задачи. В первую очередь отметим количество элементов в группах. Затем отметим наибольшее число возможных связей, и как еще одно ограничение можно учесть различные физические ограничения, которые обосновываются природой компоновываемых элементов.

Сформулируем задачу разбиения графа на куски. Пусть задан граф $G=(X,U)$, имеющий n вершин и g ребер. Требуется разбить граф на куски таким образом, чтобы получить максимальное значение функции:

$$L = \sum_{j=1}^l \sum_{i=1}^{r_j} \alpha_i^j \rightarrow \max,$$

где L общее число ребер внутри всех кусков графа.

$$\sum_{i=1}^{r_j} \alpha_i^j = \begin{cases} 1, & \text{если } i_j \in U_{jj} \\ 0, & \text{в противном случае} \end{cases},$$

где U_{jj} множество внутренних ребер куска.

Суть итерационного алгоритма заключается в выборе на каждом шаге такой пары вершин, взаимная перестановка которых приводит к уменьшению значения целевой функции L . Это будет происходить за счет перестановок строк и столбцов матрицы R (матрица смежности графа некоторой электрической схемы). Конечным результатом будут куски (G_1, G_2, \dots, G_n) , содержащие минимальное число ребер между ними,

но содержащие максимально возможное число ребер внутри каждого куска. И тем самым будет получен оптимальный вариант для дальнейшего размещения элементов на печатной плате.

Можно отметить также, что существует САПР P-CAD, с помощью которого можно осуществлять автоматическую компоновку элементов печатной платы, а именно с помощью графического редактора P-CAD Schematic и P-CAD PCB.

В заключение хотелось бы еще раз отметить важность разработки и внедрения в промышленность единого программного комплекса компоновки элементов. Данный комплекс позволит не только сократить время работы, но и повысить качество изготавливаемых плат, снизить риск появления паразитных электрических взаимосвязей и обеспечить оптимальное расположение компонентов на печатной плате.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Морозов К.К., Одинокоев В.Г., Курейчик В.М. Автоматизированное проектирование конструкций радиоэлектронной аппаратуры: Учеб. пособие для вузов. - М.: Радио и связь, 1983. - 280 с.

2. Методы разбиения схем РЭА на конструктивно законченные части/ Морозов К.К., Мелихов А.Н., Бернштейн Л.С., Одинокоев В.Г., Курейчик В.М.; под. Ред. Морозова К.К. – С.: Сов. Радио, 1978. - 136с.

УДК 620.91

И.В. САВИЛОВ, П.Е. СТРУЧКОВРязанский государственный радиотехнический университет
имени В.Ф. Уткина**АЛЬТЕРНАТИВНЫЕ ИСТОЧНИКИ ЭНЕРГИИ В
ЧАСТНОМ ДОМЕ**

В статье рассматриваются альтернативные источники энергии: ветра, сила воды, солнечная энергия, тепло земли, биотопливо, и описывается процесс их использования в частных домах для создания в будущем домов, не включенных в сеть электроснабжения.

Введение.

В окружающей среде энергия существует почти везде – земля, солнце, ветер и вода – это альтернативные ресурсы, которые преобразуются в электроэнергию.

Эффективное использование различных источников энергии в нынешних условиях разрешит на 70-85% отказаться от возможных закупок тепловой энергии, газа, и даже возможно электричества. Имеется в достаточной степени множество вариантов применения природной энергии. Необходимо произвести точный подсчет параметров энергоснабжения, взять в расчет климатический пояс, местоположение дома.



Рисунок 1 – Альтернативные источники энергии

Энергия ветра.

Применение воздушных масс в роли ветровой нагрузки предоставляет возможность достигать очень больших мощностей, около 1-15 кВт на одну вышку. Система ветряной электростанции, в общем случае включает в себя следующие составляющие:

1. Генератор;
2. Контроллер заряда;
3. Аккумуляторная батарея;
4. Инвертор.

Благодаря правильно выбранному месту для установки мачты, ветряная электростанция в качестве альтернативного источника электроэнергии окупается в течение нескольких лет. Скорость ветра возрастает с высотой. Именно поэтому электростанции такого типа устанавливают на открытых, желательнее возвышенных территориях, где есть хороший ветровой потенциал: поле, горы(холмы), а генераторы устанавливают на башнях высотой 40—60 метров. Очень важно при выборе места для размещения ветроустановки принимать во внимание предметы, которые способны снижать эффективность ветряных электростанций: крупные здания, деревья, и т. д. Главным недостатком ветроустановки остается высокий уровень шума.

Цена ветровой энергии обуславливается размером механизма, чем выше над землей установлен винт с лопастями, тем больше производительность ветряной электростанции.

Ветрогенераторы российского производства.

Разработчики ветрогенераторов в России обладают высокой квалификацией, необходимой для качественной работы. В кадрах предприятия работают специалисты высокого уровня, способные осуществить установку оборудования.

Для клиентов, имеется возможность оформить заказ для индивидуального проектирования с дальнейшим испытанием станции альтернативной добычи электроэнергии. Данную продукцию просто получить, и нет необходимость расходов на доставку товара.

Сила воды.

При наличии рядом с домом ручья или реки, поток воды можно с успехом использовать для получения электроэнергии. По запасу энергии вода уступает ветру.

Недостаточно построить только саму конструкцию водяного привода, необходимо еще построить обводной ливневый канал и запруду, что потребует не малых денежных расходов.

Гидротурбина Сара.

Представляет себе водяную турбину средних размеров, сделанную из современных смешанных материалов, которую можно погрузить в водяной поток, и ее выработанной мощности достаточно, чтобы подсоединить холодильник, телевизор и ноутбук. Японские инженеры из компании Ibasei считают, что выпустив в прошлом году крайнюю разработку - гидротурбину Сарра, они совершили большой прорыв.

Турбина должна быть установлена в поток воды на специальных креплениях. При скорости потока в 1,9 м/сек, эта система может давать на выходе до 250 Вт мощности.

Как утверждают представители компании – в существе используется диффузор необычной формы, в результате чего небольшой поток воды ускоряется, и крутит лопасти, в результате чего вырабатывается электрический ток.

По расчетам специалистов, при цене турбины в порядке 10000\$ она станет незаменимой установкой в добыче электроэнергии в частных домах.

Энергия солнца и кремниевые панели.

Самым распространенным на сегодняшний день альтернативным источником является солнечная энергия. Солнечные батареи являются самыми бесшумными и безопасными для природы источниками энергии. Перед тем как приобретать солнечные панели и устанавливать их в роли основного источника энергии, нужно не забывать о некоторых минусах данного способа получения электроэнергии:

1. Большая цена;
2. Низкая мощность
3. Сложность ремонта.

Вышеуказанные недостатки в основном определяются высокой ценой солнечного элемента. По мнению экспертов, снижение цены на кремневые батареи всего на 60% приведет к большому спросу на данный источник энергии.

Не мало важным преимуществом солнечных панелей является то, что для установки их на крыше частного дома не нужны никакие разрешения властей.

Sun Power – небольшого размера солнечные панели.

Предприятие Sun Power располагается в Украине. Стала известной выпускаемыми портативными солнечными плитами. Их используют, чтобы получить электричество даже в таких условиях, как походных. Они отличны от остальных своей мобильностью, средними размерами. Способны вырабатывать мощность до 500 Вт.

В таблице 1 рассматриваются сравнительные характеристики некоторых изделий.

Таблица 1.

Изделие	Вес	Мощность	Стоимость
Портативная солнечная электростанция	18 кг	500 Вт	40 000 руб.
Блок питания на солнечных модулях	До 3 кг	До 50 Вт	3 000 руб.
Сборочные солнечные электростанции	—	От 1 кВт до 11 кВт и выше	От 1 699 у. е. за 1 кВт

Тепло земли.

Тепло земли является самым надежным источником энергии. Тепловые насосы получают энергию из земли или грунтовых вод, тем самым позволяя сократить расходы на отопление. Тепловые насосы относятся к одной из самых эффективных систем естественных источников энергии. Тепловой насос расходует на 50% меньше тепловой энергии, чем потребляет.

Для получения потока с температурой 60-700 С, необходимо поместить трубы на глубину 5-6 метров ниже уровня промерзания грунта. Тепло внутренних слоев грунта передается в тепловой насос, который использует это тепло для обогрева помещений.

По внешнему источнику тепла тепловые насосы разделяются на 3 основных типа:

1. Земляные,
2. Водяные,
3. Воздушные;

Использование тепловых насосов и стоимость их подключения представлено в таблице 2.

Таблица 2.

Тепловые насосы для систем отопления и горячего водоснабжения		
WATERKOTTE EcoTouch DS 5027.5 Ai (Германия)	Происходит отбор тепла от почвы и грунтовых вод; так же есть возможность эксплуатации содержащих каскадные системы; способен работать от трёхфазной сети.	708 521р

DANFOSS DHP-R ECO 42 (Швеция) 9.6= 42 65 380	Управление мощностью компрессора и частотой вращения циркуляционных насосов посредством регулировки частоты; дополнительный теплообменник; сеть – 380 V.	1 180 453р.
Viessmann Vitocal 300-G WWC 110 (Германия)	Насосы первичного и вторичного контура; имеется возможность подключения гелиосистем.	630 125р

Биотопливо.

Большинство альтернативных источников энергии обладают одним и тем же недостатком – сделать запас тепла или электроэнергии практически невозможно, если не считать сверхбольших теплоаккумуляторов и маломощных литиевых батарей к солнечным панелям.

Большинство владельцев частных домов предпочли бы использовать альтернативный вариант безопасного и простого в использовании биотоплива, которое можно было бы запастись на весь отопительный сезон.

На сегодня используются два варианта альтернативного топлива:

- Биогаз, получаемый непосредственно на территории усадьбы или домовладения;
- Пеллеты, гранулированные продукты переработки угля, торфа, древесины, отходов лесопиления.

Источником сырья для производства пеллет могут быть любые отходы древесины. Небольшой ротационный пресс, который можно легко установить в домашних условиях, превращает мешок измельченной стружки в несколько килограммов пеллет. В результате владелец получает источник альтернативного недорогого топлива, которое можно запастись и сжигать в специальных котлах с автоматической подачей пеллетной массы.

Удобный, но не самый безопасный источник альтернативной энергии, так как биогаз не имеет запаха и в случае утечки может легко привести к возникновению пожара.

Вывод: в данной работе, были рассмотрены альтернативные источники электропитания для частного дома, где главное питание происходило путем подключения к основной сети, а дополнительным были рассмотрены некоторые другие источники, которые, в будущем, смогут заменить систему электропитания домов.

Можно заметить, что из рассмотренных выше альтернативных источников, все способны работать надежно, так и в аварийном режиме.

В заключении, можно подвести итоги и предложения по дальнейшему использованию альтернативных источников электроэнергии, как дополнительных, увеличивающих энергоэкономичность и продуктивность при строительстве зданий и сооружений, а так же окупаемость проекта и его дальнейшее развитие в условиях центрального региона России.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Шефтер Я.И. Использование энергии ветра. // Шефтер Я.И. // Энергоатомиздат. Москва. 1983г. Дата обращения 7.03.2021г.

2. Альтернативные источники энергии для частного дома [Электронный ресурс]. Режим доступа: <https://zen.yandex.ru/media/id/5c3705e153bb6f00aaaf22fd/alternativnye-istochniki-energii-dlia-hastnogodoma5c3757f4ed659f00ac982e2d/> Дата обращения 8.03.2021.

3. Вольфганг Ф. Основные положения по проектированию пассивных домов / Вольфганг Ф. // – АСВ, 2011. – 145 с

4. Гоньшаков А.Г., Власов А.В., Лисятников М.С. Учеб. пособие “Особенности проектирования пассивного дома” / Гоньшаков А.Г., Власов А.В., Лисятников М.С. // – 2014. – 112 с.

5. Гибридные системы для энергоснабжения дома [Электронный ресурс] Режим доступа: <https://alterair.ua/articles/gibridnyie-sistemyi-dlya-doma/> Дата обращения 6.03.2021

6. Солнечная энергетика России: перспективы и проблемы развития [Электронный ресурс] Режим доступа: <https://gisee.ru/audit/articles/solar-energy/24510/> Дата обращения 6.03.2021.

УДК 004.032.26

П.А. САЛЫНОВ, С.В. СКВОРЦОВРязанский государственный радиотехнический университет
имени В.Ф. Уткина**ПРИМЕНЕНИЕ РЕКУРРЕНТНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ
РЕШЕНИЯ ЗАДАЧИ РАСПОЗНАВАНИЯ РЕЧИ***Рассматриваются принципы работы нейронных сетей с рекуррентной архитектурой на примере модели LSTM и её применение в задаче распознавания речи.*

Задача машинного распознавания речи в настоящее время является одной из наиболее востребованных в различных областях науки и техники и решается при помощи технологий нейронных сетей. Однако эту задачу невозможно решить при использовании простых нейронных сетей с прямым распространением сигнала из-за 2 основных факторов: 1) обрабатываемые данные имеют неопределённую длину/размер; 2) обрабатываемые данные взаимосвязаны между собой. При решении задач обработки данных неопределённой длины или размера они предварительно декомпозируются или обрабатываются специальным алгоритмом для того, чтобы входные данные имели структуру, подходящую для конкретной нейронной сети. Однако не всегда это получается сделать без потери взаимосвязи данных.

Так для решения задачи распознавания речи требуется определить последовательность лексем, проанализировав входной звуковой сигнал. Каждая из лексем может принимать неопределённую длину в самом звуковом сигнале, поэтому для решения данной задачи необходимо каким-либо образом определить временные промежутки, внутри которых находится каждая лексема. Однако зачастую сделать правильно это либо невозможно, либо крайне сложно.

Для решения подобных задач была разработана концепция рекуррентных нейронных сетей (RNN, Recurrent neural network)[1]. Основная идея концепции заключается в хранении определённого внутреннего состояния в нейронной сети, которое будет меняться в зависимости от передаваемого входного сигнала и влиять на значение выходного. На рисунке 1 приведён пример схемы рекуррентной нейронной сети, развёрнутой во времени.

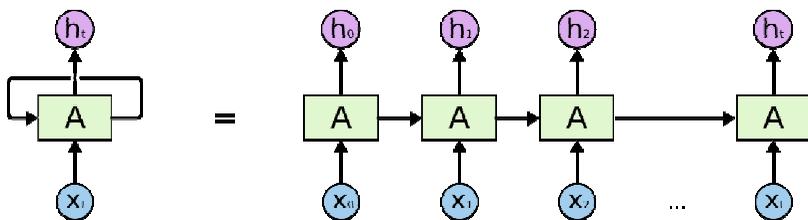


Рисунок 1 – Концепция структуры рекуррентных нейронных сетей, развёрнутая во времени

Такая модель позволяет разделить весь поток входных данных на определённые части, но при этом обрабатывать их как цельный поток, при этом сохраняя общую информацию обо всех проделанных до этого операциях во внутреннем состоянии. Самой популярной архитектурой рекуррентной нейронной сети на текущий момент времени является LSTM.

LSTM (Long short-term memory) – разновидность архитектуры рекуррентных нейронных сетей, предложенная в 1997 году Зеппом Хохрайтером и Юргеном Шмидхубером[2]. Её главная особенность заключается в том, что она способна сохранять внутреннее состояние на длительные промежутки времени.

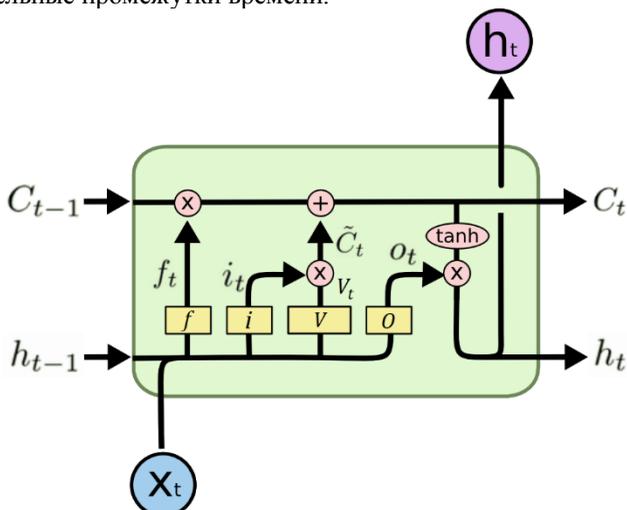


Рисунок 2 – структура рекуррентной нейронной сети LSTM

Как видно из рисунка, LSTM-сеть имеет три входных сигнала (x_t , h_{t-1} , C_{t-1}) и два выходных (h_t , C_t), каждый из которых

представляет собой вектор значений, где, C_t – внутреннее состояние нейронной сети в момент времени t ; h_t – значение на выходном слое нейронной сети в момент времени t ; x_t – значение на входном слое нейронной сети в момент времени t .

Для изменения внутреннего состояния C применяются так называемые вентили (gate)[3], каждый из которых состоит из одного или двух слоёв нейронов сети (обозначены прямоугольниками).

Всего в LSTM структуре используется три типа вентиляй: вентиль забвения, входной вентиль, выходной вентиль.

Вентиль забвения производит поэлементное умножение значений вектора состояния C_{t-1} и вектора f_t , который вычисляется по следующей формуле:

$$f_t = \sigma(W_f h_{t-1} + W_f x_t + b_f), \quad (1)$$

где W_f – матрица весов слоя f , b_f – вектор смещения слоя f .

Вентиль забвения используется для полного или частичного обнуления переменных состояния и принимает значения в диапазоне $[0,1]$, причем 0 означает полное стирание переменной, а 1 – передачу в неизменённом виде.

Входной вентиль используется для записи новых значений во внутреннее состояние сети. Он производит поэлементное сложение значений вектора состояний C_{t-1} и вектора \tilde{C}_t , который вычисляется по следующей формуле:

$$\tilde{C}_t = i_t V_t, \quad (2)$$

где V_t – выходной вектор нейронного слоя с функцией активации гиперболический тангенс; i_t – выходной вектор нейронного слоя с функцией активации сигмоида.

Вектор V_t представляет собой сами данные, которые требуется записать в состояние C_t . V_t вычисляется по следующей формуле:

$$V_t = \tanh(W_v h_{t-1} + W_v x_t + b_v), \quad (3)$$

где W_v – матрица весов слоя V , b_v – вектор смещения слоя V .

Вектор i_t определяет степень необходимости записи каждого значения вектора \tilde{C}_t , где значение 0 – не выполнение записи, 1 – запись значения в неизменном виде. Он вычисляется по следующей формуле:

$$i_t = \sigma(W_i h_{t-1} + W_i x_t + b_i), \quad (4)$$

где W_i – матрица весов слоя i ; b_i – вектор смещения слоя i .

Выходной вентиль формирует выходной вектор \hat{h}_t с учётом текущего изменённого внутреннего состояния C_t и входного вектора x_t . Вектор \hat{h}_t вычисляется по следующей формуле:

$$\hat{h}_t = o_t \tanh(C_t), \quad (5)$$

где o_t – представляет собой выходной вектор значений нейронного слоя с функцией активации сигмоида. Он вычисляется как

$$o_t = \sigma(W_o h_{t-1} + W_o x_t + b_o), \quad (6)$$

где W_o – матрица весов слоя o , b_o – вектор смещения слоя o .

Несмотря на сложную структуру сетей LSTM они могут обучаться теми же алгоритмами, что и сети с прямым распространением сигнала, например, методом обратного распространения ошибки.

При практическом применении в задаче распознавания речи данная структура показала хорошие результаты. Структура LSTM была основным компонентом сети, которая в 2013 году достигла рекордного порога ошибки в 17,7 % в задаче распознавания фонов на классическом корпусе естественной речи ТИМТ[4].

Таким образом рекуррентная структура нейронных сетей стали большим шагом вперёд в решении задач распознавания речи и стала главным архитектурным стандартом, используемым во всех сетях, применяемых в задачах распознавания речи.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Sak, Hasim Long Short-Term Memory recurrent neural network architectures for large scale acoustic modeling, 2014.
2. Sepp Hochreiter; Jürgen Schmidhuber. Long short-term memory, 1997 с. 1735—1780
3. Understanding LSTM Networks [Электронный ресурс]. – Режим доступа: <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>.
4. Graves, Alex; Mohamed, Abdel-rahman; Hinton, Geoffrey. Speech Recognition with Deep Recurrent Neural Networks, 2013 с. 6645—6649.

УДК 621.396.99

А.В. СИЛКИНМежвидовой центр подготовки и боевого применения войск РЭБ
(учебный и испытательный)**ПРИМЕНЕНИЕ РАДИОУСТРОЙСТВ С
ПРОГРАММИРУЕМЫМИ ПАРАМЕТРАМИ В
РАДИОТЕХНИЧЕСКИХ СИСТЕМАХ**

Рассматривается возможное применение «любительских» проектов, созданных на основе технологии Software Defined Radio (SDR), как основы для создания радиотехнических систем разведки.

Основной задачей радиоэлектронной защиты (РЭЗ) является устранение или ослабление воздействия на свои радиоэлектронные объекты средств радиоэлектронного поражения противника. Одним из способов осуществления мероприятий РЭЗ, например, в области радиолокации, является переход от активных систем к пассивным, в том числе многопозиционным. Главной особенностью работы таких систем является скрытность и, как следствие, защищенность по отношению к средствам разведки и радиоэлектронной борьбы (РЭБ) противника, что соответственно приводит к повышению живучести в условиях современного боя.

В качестве источников информации об объектах наблюдения являются регистрируемые приемными позициями сигналы, излучаемые непосредственно объектами, или отраженные от объектов сигналы подсвета, излучаемые внешними источниками. Чаще всего в качестве подсвета используют сигналы широкоэвещательных телевизионных (ТВ) или ультракоротковолновых (УКВ) радиопередатчиков, а также станций сотовой связи, спутниковых навигационных систем и т.д.

Отметим, что прием и анализ отраженных сигналов подсвета является источником информации об объектах не только в радиолокации, но и расширяет возможности радиоэлектронной разведки.

По данным открытых источников [1], большое внимание и усилия в странах НАТО уделялось и уделяется разработке и созданию макетов и систем пассивной многопозиционной радиолокации. Одной из таких систем стала Silent Sentry 2, разработанная компанией Lockheed Martin и использующая в качестве сигналов подсвета излучение ТВ и УКВ радиопередатчиков, а разработанная компаний

Roke Manor система CELLDAR использует сотовую связь GSM 900, 1800 и 3G. Также, в интересах армии США для решения задач противовоздушной обороны в условиях РЭБ в зонах конфликтов компанией Raytheon разработана система Homeland Alert 100. Ее предназначение – получение данных о малоскоростных маловысотных воздушных целях, в том числе беспилотных летательных аппаратов, крылатых ракет и целях, созданных по технологии "стелс". Сигналами подсвета, как и в Silent Sentry, является ТВ и УКВ радиопередатчики.

Достоинством подобных систем является меньшая стоимость производства, размещения и эксплуатации, отсутствие помех своим радиоэлектронным средствам, т.е. обеспечение вопроса электромагнитной совместимости. Среди недостатков – отсутствие контроля над передающей частью системы. Неизвестным могут быть ее положение, тип сигнала и режим работы, отсутствует возможность их изменять.

В условиях современных локальных конфликтов и широкомасштабных боевых действий применение описанных ранее систем затруднительно из-за полного или частичного отсутствия источников подсвета в виде гражданских систем. Однако отметим, что в ходе военных действий противоборствующие стороны развертывают различные системы управления войсками и оружием. Так, на рисунке 1 представлена частотная панорама диапазона от 415 МГц до 475 МГц, сделанная в ходе вооруженного конфликта в одной из Ближневосточных Стран. Большая часть обнаруженных в ходе мониторинга работающих частот относится к мощным ретрансляторам транковой радиосвязи стандарта DMR, которые нашли широкое применение у представителей незаконных вооруженных формирований. Информация об использовании сигналов подобных систем в качестве подсвета в открытой печати в настоящее время отсутствует.

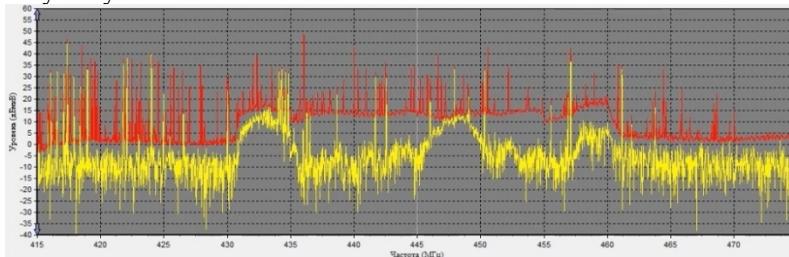


Рисунок 1 – Частотная панорама

Важнейшей частью рассматриваемых систем является устройство обработки принимаемых сигналов, причем тенденцией

последнего времени является переход к концепциям и технологиям SDR. По определению МСЭ [2], SDR – радиопередатчик и/или радиоприемник, использующий технологию, позволяющую с помощью программного обеспечения устанавливать или изменять рабочие радиочастотные параметры, включая, в частности, диапазон частот, тип модуляции или выходную мощность, за исключением изменения рабочих параметров, используемых в ходе обычной, предварительно определенной работы с предварительными установками радиоустройства, согласно той или иной спецификации или стандарта системы.

Отметим, что свое применение технология SDR нашла давно, например, при реализации системы SPEAKEasy [3], созданной для армии США. Но именно появление на рынке за последнее десятилетие и широкое развитие программных и аппаратных доступных «любительских» платформ (таблица 1), сделало технологию SDR доступной для любых пользователей. В этом и заключается главная особенность – платформы являются полностью открытыми, совершенно бесплатно доступны схемы устройств и печатных плат, исходный код программ и микроконтроллеров.

Таблица 1 – «Любительские» платформы SDR

Наименование	HackRF One	BladeRF	LimeSDR (рисунок 2)	RTL-SDR
Диапазон рабочих частот	1 МГц – 6 ГГц	300 МГц – 3,8 ГГц	100 кГц – 3,8 ГГц	22 МГц – 2,2 ГГц
Полоса рабочих частот	20 МГц	40 МГц	61,44 МГц	3,2 МГц
Разрядность АЦП/ЦАП	8 бит	12 бит	12 бит	8 бит
Частота дискретизации	20 МГц	40 МГц	61,44 МГц	3,2 МГц
Цена	299\$	420\$	299\$	~10\$



Рисунок 2 – LimeSDR

Самой доступной «любительской» платформой для работы с SDR является RTL-SDR, в основе работы которой лежит ТВ-тюнер

DVB-T на чипе RTL2832U. Данное устройство нашло широкое применение у ученых, исследователей и пользователей по всему миру. Так, например [4], RTL-SDR является основой проектов KerberosSDR и KrakenSDR – многоканальных пассивных системам, использующих ТВ и УКВ гражданские радиопередатчики для наблюдения за транспортными средствами и самолетами, т.е. в какой-то степени выполняющими функции военных систем рассмотренных ранее.

На рисунке 3 показан пример работы проекта KrakenSDR. Красная линия – фактические координаты самолета, полученные по данным ADS-B, черные линии – измерения по сигналу DAB, отраженному от корпуса самолета.



Рисунок 3 – Работа проекта KrakenSDR

Доступность в настоящее время технологии SDR пользователям по всему миру, открытость разрабатываемых ими проектов и способов применения технологии – источник идей и решений для создания в интересах Вооруженных сил любой из стран систем и средств радиоэлектронной разведки, например, малогабаритных радиотехнических средств разведки для оснащения маневренных групп.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Петров В., Гришулин С. Наземные радиолокационные станции ПВО-ПРО на ТВД стран НАТО. Зарубежное военное обозрение. 2010. №9. С. 54-58.
2. Отчет МСЭ-R SM.2152. Определения системы радиосвязи с программируемыми параметрами (SDR) и системы когнитивного радио (CRS). 2010.
3. Cook P., Bonser W. Architectural overview of the SPEAKeasy system. IEEE Journal on Selected Areas in Communications. 1999. №4. P. 650-651.
4. Сайт платформы RTL-SDR <http://www.rtl-sdr.com>

УДК 004.03

А.Ю. СТЕБЛЮКРязанский государственный радиотехнический университет
имени В.Ф. Уткина**ОСОБЕННОСТИ ОРГАНИЗАЦИИ И ФУНКЦИОНИРОВАНИЯ
ИНФОРМАЦИОННО-АНАЛИТИЧЕСКИХ СИСТЕМ**

В работе рассматриваются особенности организации и функционирования информационно-аналитических систем, указываются их основные составляющие, а так же отмечаются возможные сферы их применения.

В современном мире, где информационные технологии совершили значительный скачек, возникла потребность в контроле и мониторинге больших объемов данных, циркулирующих в рамках работы какого-либо предприятия, обрабатывать которые вручную человек не в силах. Поэтому инженерами были разработаны специализированные системы для контроля и мониторинга процессов и потоков данных, протекающих в производственном предприятии. Разновидностей таких систем существует широкое множество, в данной работе следует обратить особое внимание на информационно-аналитические системы.

Информационно-аналитические системы представляет собой отдельный тип информационных систем, предназначенных для анализа и последующей обработки потоков данных. Основная задача таких систем заключается в осуществлении поиска, сбора, последующей обработки и представлении полученных данных в виде, пригодном для восприятия, анализа и применения оператором такой системы [1]. Заявленный функционал позволяет организовать оптимальный управленческий или иного рода процесс, протекающий в рамках конкретного предприятия.

Говоря об уровнях, на которых осуществляется функционирование информационно-аналитических систем, можно отметить следующие:

1. Уровень обнаружения, сбора, хранения и передачи информации;
2. Уровень аналитики, на котором осуществляется обработка поступающей информации, а именно её обобщение, сортировка и выполнения анализа, в результате которого формируются выводы с последующим составлением прогнозов.

В соответствии с отмеченными уровнями современные информационно-аналитические системы, служащие для обеспечения специалистов данными о различного рода вопросах и проблемах в работе предприятия, состоят из определенного набора подсистем. В классическом варианте реализации информационно-аналитической системы, выделяют следующие две её составляющих:

1. Подсистема, предназначенная для сбора и хранения информации о предприятии. Она осуществляет сбор и фильтрацию полученных данных, а также последующее накопление информации, обеспечивая её оперативное применение для предстоящих аналитических операций [4, 5, 6].

2. Подсистема, отвечающая за доступ к информации, её дальнейший анализ и составление отчетности обслуживаемой организации [3]. Обычно содержит в своём составе блоки, обеспечивающие доступ к требуемой информации, её безопасность и своевременную блокировку несанкционированного доступа к ней.

Подсистемы, принадлежащие к первому типу, основаны на технологии оперативной обработки транзакций, которая содержит в своей основе концепцию хранилища данных. Хранилище данных служит для хранения достоверных обобщенных данных о предприятии, отражающих его работу за длительный временной промежуток, а также информацию о внешней среде его деятельности. Зачастую, вместе с хранилищем информации, для достижения положительного эффекта применяют информационные витрины, представляющие собой специальные базы аналитики данных, организованные на основе одного из отдельных подразделений или направлений работы предприятия.

Аналитическая пирамида.

Ранее упомянутые составляющие подсистем могут быть отражены в «аналитической пирамиде», представляющую собой информационную инфраструктуру (рисунок 1):



Рисунок 1 – Аналитическая пирамида

Аналитическая пирамида, по своей сути, представляет собой организованную определенным образом иерархическую структуру, в которой различные классы информационных систем отнесены на отдельные уровни. В её основании лежат транзакционные системы, принципы которых заложен в суть работы и функционирования информационно-аналитических систем. По мере движения от основания пирамиды к её вершине происходит агрегирование обрабатываемых данных в информацию, предназначенную для поддержки принятия управленческих решений.

Возвращаясь к теме информационно-аналитической систем, важно подчеркнуть похожую тенденцию с «аналитической пирамидой» - главной задачей формирования таких систем является передвижение информационной базы предприятия из неопределенной среды в логическим образом организованную систему, являющуюся информационным источником, который способен удовлетворить основные нужды и потребности предприятия [2, 3].

Как говорилось ранее, в основе базиса информационно-аналитических систем, как лежит хранилище информационных данных, которое:

1. Считается информационной базой, служащей лишь для чтения;
2. Подвергается обновлению из операционных систем на постоянной основе;
3. Предполагает один подход к обозначению и сохранению информации, независимо от её организации.

В качестве прикладного примера приведем схему структуры некоего абстрактного предприятия (рисунок 2):

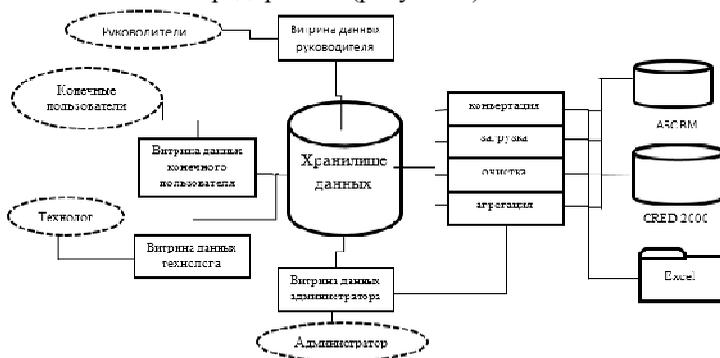


Рисунок 2 – Структурная схема информационно-аналитической системы

Как видно из приведенного рисунка, в данной системе используется несколько витрин данных, которые обеспечивают представление аналитических данных для каждого конкретного вида пользователей, будь-то руководитель предприятия, администратор или конечный пользователь, например, обычный оператор данной системы. В центре схемы расположено хранилище данных, служащее для накопления и безопасного хранения общих данных предприятия, полученных в процессе его работы.

Интерфейс информационно-аналитических систем.

Также, вспоминая основные функции информационно-аналитических систем, стоит подробнее остановиться на их способности корректного отображения данных для пользователя данной системы.

Для обеспечения такой возможности, помимо использования витрин данных для каждого вида пользователей, следует грамотно подойти к проектированию интерфейса данных систем. С точки зрения представления информации главным принципом должно являться абстрагирование от лишних подробностей обработки данных и работы системы, а также вывод нужной информации в форме, доступной для восприятия пользователя.

Что касается последнего, учеными были выявлены некоторые принципы проектирования визуального интерфейса пользователя, которые позволяют достичь наибольшей наглядности представления информации, что приводит к приросту производительности работы пользователя такой системы.

Основными принципами проектирования интерфейсов систем являются:

1. Принцип «Золотого сечения»;

Данный принцип заключается в правильных пропорциях информационных блоков, что повышает наглядность представляемой информации.

2. Кошелек Миллера;

Этот принцип гласит, что в дизайне интерфейсов стоит группировать элементы блоков не более, чем семь элементов в группе.

3. Принцип группировки;

Согласно этому правилу, интерфейс программы должен быть разбит на ясно очерченные блоки элементов с заголовком для каждого.

4. Бритва Оккама и др.

Перечисленные принципы проектирования интерфейсов являются наиболее широко используемыми в современных прикладных инженерных решениях. С помощью них можно достичь

значительного прироста в продуктивности работы оператора той или иной системы, что, в свою очередь, повысит качество принимаемых решений по дальнейшей политике функционирования предприятия и увеличит его прибыль.

Заключение.

Таким образом, информационно-аналитические системы позволяют значительно упростить процесс мониторинга и контроля информации в течении работы производственного предприятия, а грамотно спроектированный интерфейс данных систем дает значительное увеличение качества работы и продуктивности оператора информационно-аналитической системы. Итогом проведенной работы является разработанный интерфейс для информационно-аналитической системы по контролю за природными ресурсами и выбросами в атмосферу.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Исаев Д.В. Системы бизнес-интеллекта и аналитические приложения. [Текст] - Финансовая газета, 2005, №31 (711). – с.14-15, №32 (712). – с.15.

2. Кравченко Т.К., Пресняков В.Ф. Инфокоммуникационные технологии управления предприятием. [Текст] – М.: ГУ-ВШЭ, 2003. – 272 с.

3. Лодон Дж., Лодон К. Управление информационными системами / Пер.с англ. под ред. Д.Р.Трутнева. [Текст] – СПб.: Питер, 2005. – 912 с.

4. Исаев Д.В. Аналитические информационные системы. [Текст] – М.: ГУ-ВШЭ, 2008. – 60 с.

5. Каплан Р., Нортон Д. Стратегическое единство. Создание синергии организации с помощью сбалансированной системы показателей. [Текст] – М.: Вильямс, 2006. – 384 с.

6. Информационно-аналитические системы: актуальность и примеры [Электронный ресурс] – URL: https://spravochnick.ru/informatika/informacionno-analiticheskie_sistemy_aktualnost_i_primery/

УДК 004.021

А.С. УШАНОВРязанский государственный радиотехнический университет
имени В.Ф. Уткина**СТЕРЕОЗРЕНИЕ В ЗАДАЧАХ ОБНАРУЖЕНИЯ И
ТРАЕКТОРНОГО СОПРОВОЖДЕНИЯ ОБЪЕКТОВ**

Рассматривается алгоритм использования стереозрения в задачах обнаружения и траекторного сопровождения для получения трёхмерных координат отслеживаемых объектов.

Задача обнаружения и траекторного сопровождения объектов в настоящее время является одной из основных в области компьютерного зрения. Она заключается в нахождении на изображении объекта, обладающего некоторыми определенными характеристиками.

Для обнаружения на изображении наличия одного или нескольких объектов чаще всего используется метод вычитания фона. Для этого берется некоторое статичное нейтральное изображение фона, на котором нет объектов. Дальше из каждого кадра видеопотока вычитается исходный кадр фона, полученная разница переводится в бинарный вид и используется в качестве маски для выделения объектов, отличных от фона.

Данный метод используется в простых задачах, где не важна точность положения обнаруживаемого объекта. Если необходимо получить точные пространственные координаты, то может помочь использование стерео зрения. В качестве приемников используются две откалиброванные видеокамеры, разнесенные в пространстве друг от друга. Калибровка видеокамер означает, что известна матрица преобразований, которая связывает координаты точек изображения с трёхмерными мировыми координатами. Для каждой отдельной камеры производится обнаружение объекта в видеопотоке и определение его двухмерных координат. С помощью эпиполярной геометрии [2] определяется соответствие между координатами объекта, обнаруженного первой камерой и координатами объекта, обнаруженного второй камерой (Рисунок 1). Используются две камеры, где C – центр первой камеры, аналогично C' – центр второй камеры. Некоторая точка трёхмерного пространства X проецируется в x на плоскость изображения одной из имеющихся камер. Тогда прообразом точки x на изображении второй камеры является луч xX .

Луч проецируется на плоскость второй имеющейся камеры в прямую l' , которая называется эпиполярной линией. Образ точки X на плоскости изображения второй камеры обязательно лежит на эпиполярной линии. Данная геометрия используется для нахождения стереопар и проверки, могут ли две точки быть стереопарой [3].

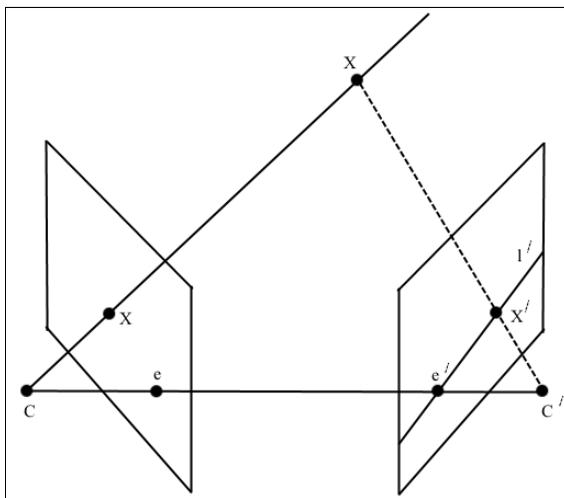


Рисунок 1 – Эпиполярная геометрия

Зная координаты соответствующих точек траектории движения объекта на изображениях, полученных с видеокамер, и используя матрицу преобразований, можно определить трёхмерную мировую координату точки траектории движения объекта.

Алгоритм определения трёхмерных координат объекта можно разделить на следующие этапы:

1. Определение внутренних параметров камер (калибровка камер).
2. Определение внешних параметров камер (привязка камер).
3. Обнаружение объекта на кадрах видеопотока и определение соответствующих координат, полученных с разных камер.
4. Определение трёхмерных координат объекта.

Процедура калибровки камер описана в [1]. Для определения трёхмерных координат обычно используется триангуляция точек.

Пусть имеются две откалиброванные камеры, а так же однородные координаты проекций объекта. Тогда составляется система уравнений:

$$\begin{cases} x_1 = P_1 X \\ x_2 = P_2 X \end{cases}$$

где x_1, x_2 – однородные координаты проекций объекта пространства X , P_1, P_2 – матрицы преобразований камер. Далее решается система этих уравнений.

Данный метод позволяет соотнести координаты обнаруженного видеокameraми объекта с мировыми трёхмерными координатами. Систему из двух камер можно использовать на охраняемых территориях, где важна точность обнаружения движущихся объектов для предотвращения их пересечения определённых границ, так же заданных в трёхмерном пространстве.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Z. Zhang, «A flexible new technique for camera calibration», IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.22, No.11, pages 1330-1334, 2000.

2. Грузман И.С., Киричук В.С., Косых В.П., Перетягин Г.И., Спектор А.А. ЦИФРОВАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ В ИНФОРМАЦИОННЫХ СИСТЕМАХ: Учебное пособие.- Новосибирск: Изд-во НГТУ, 2000.

3. Основы стереозрения [Электронный ресурс]. – URL: <https://habr.com/ru/post/130300> (дата обращения: 09.04.2021)

4. Z. Kim, "Geometry of Vanishing Points and its Application to External Calibration and Realtime Pose Estimation", Institute of Transportation Studies, Research Reports, Paper UCB-ITS-RR-2006-5, July 1, 2006.

УДК 621.311.62

К.Ф. ФИЛЬКИНРязанский государственный радиотехнический университет
имени В.Ф. Уткина**ОБЕСПЕЧЕНИЕ ЭЛЕКТРОМАГНИТНОЙ СОВМЕСТИМОСТИ
КОРРЕКТОРА КОЭФФИЦИЕНТА МОЩНОСТИ ВТОРИЧНОГО
ИСТОЧНИКА ЭЛЕКТРОПИТАНИЯ**

Рассматриваются основы учета и способы обеспечения электромагнитной совместимости (далее по тексту – ЭМС) корректора коэффициента мощности (далее по тексту - ККМ) на этапе проектирования.

Введение. Вторичный источник электропитания является промежуточным основополагающим звеном между системой электроснабжения (первичными питающими структурами) и потребителем электрической энергии. Увеличение частоты работы преобразователя, предъявляемые требования по наличию ККМ обостряют необходимость применения мер по устранению влияния электромагнитного излучения как во внутренних узлах оборудования, так и обеспечения его устойчивости от внешних воздействий.

Помехи, воздействию которых подвержен ККМ. На этапе проектирования секции ККМ разработчик должен учитывать:

- а) Наводки со стороны сети переменного электрического тока.
- б) Внутриблоковые помехи, наводящиеся ввиду функциональных особенностей работы элементов самого ККМ.
- в) Помехи на выходной линии передачи энергии в нагрузку.

На рисунке 1 приведен наглядный пример распространения кондуктивной помехи и путь ее замыкания от источника переменного электрического тока во вторичную электрическую цепь при отсутствии защитных мер разработчика:

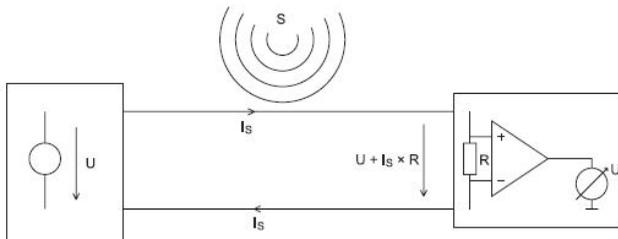


Рисунок 1 – Распространение и путь замыкания противофазной кондуктивной помехи

На рисунке 2 в качестве S обозначена помеха, в качестве I_s – ток помехи.

Вопрос затрат для обеспечения электромагнитной совместимости. На рисунке 2 приведен график анализа объема затрат по обеспечению электромагнитной совместимости ККМ на различных этапах жизненного цикла изделия. Согласно графику можно понять, что обеспечение ЭМС ККМ разработчиком наиболее рационально на этапах предпроектного исследования и проектирования с точки зрения эффективности применяемых мер в целом, а также в части экономичности. На этапе единичного или серийного производства эффективность решения вопроса по ЭМС значительно падает, что показывает кривая пунктирная линия на рисунке 2. Потери времени на доработку оказывают негативное влияние на экономическую составляющую и реализацию устройства в качестве товара на рынке.

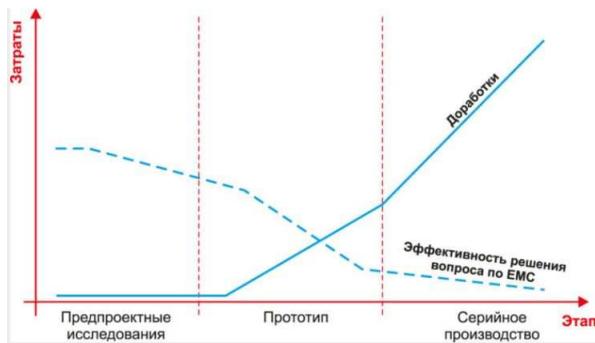


Рисунок 2 – График затрат по решению вопроса ЭМС в ККМ на основных этапах

Меры, учитываемые разработчиком при обеспечении ЭМС ККМ. Для устранения собственных помех ККМ, разработчику следует применять в качестве элементной силовой базы высококачественные компоненты, позволяющие обеспечить требуемое быстродействие (переключающие полупроводниковые приборы), различные цепи ограничения нарастания напряжения (демпферы и т.д.), а также оптимизация площади силовых контуров с большими токами. На этапе компоновки и трассировки печатной платы, к проведению печатных проводников необходимо приступать только после оптимальной компоновки узлов и компонентов (с учетом критерия минимизации взаимного влияния).

Для устранения сетевых помех от сети переменного электрического тока, самым общим и минимальным способом обеспечения защиты является применение сетевого помехоподавляющего фильтра, включающего различные защиты (подавления скачков напряжения, симметричных и несимметричных сетевых помех), позволяющие ограничить распространение наведенной помехи как в ККМ, так и обратно в сеть переменного электрического тока.

Для устранения помехи на выходной линии ККМ следует обращать внимание на особенности схмотехнического разнесения управляющих и силовых узлов и оптимальной разводки цепей заземления с целью ограничения импульсных помех, а также применение высокочастотных индуктивных компонентов (дресселей) с малой паразитной емкостью.

Особое внимание при учете влияния помех излучения на ККМ из внешней среды и внутреннего излучения самого ККМ следует уделить экранированию корпуса устройства, а также экранирования силовых индуктивных компонентов.

Заключение. При пренебрежении обеспечения ЭМС в силовой электронике (в частности, ККМ) разработчики рискуют столкнуться с появлением недопустимого уровня помех в непредвиденных аспектах (режимы функционирования, диапазонах частот). Именно поэтому в техническом задании, технических условиях на изделие обязательно предъявляются требования по обеспечению ЭМС.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Браун, М. Источники питания. Расчет и конструирование.: Пер. с англ. – К.: «МК-Пресс», 2007. – 288 с., ил.
2. Мэк, Р. Импульсные источники питания. Теоретические основы проектирования и руководство по практическому применению / Пер. С англ / Р. Мэк. – М.: Издательский дом «Додэка – XXI», 2008. - 272 с.
3. ККМ [Электронный ресурс]. Режим доступа: <https://khomovelectro.ru/korreksii-koeffitsienta-moshchnosti.html>

УДК 621.311.62

К.Ф. ФИЛЬКИН

Рязанский государственный радиотехнический университет
имени В.Ф. Уткина

ПРОГРАММНЫЙ ПОДХОД РЕАЛИЗАЦИИ АЛГОРИТМА УПРАВЛЕНИЯ ККМ НА ОСНОВЕ МИКРОКОНТРОЛЛЕРОВ

Рассматриваются основные способы обеспечения цифрового управления силовой частью корректора коэффициента мощности (далее по тексту - ККМ) и реализации защитных функций программным способом.

Введение. По европейским стандартам электромагнитной совместимости все импульсные источники электропитания мощностью более 18 Вт должны иметь ККМ. Сложность источников электропитания с ККМ обусловлена растущими к ним требованиями. В основном это связано с двумя необходимостями:

- а) Обеспечение более высокой эффективности вторичных импульсных источников электропитания и требованиями к качеству потребляемой от сети энергии переменного электрического тока.
- б) Меньшими размерами, весом и стоимостью.

Основной алгоритм управления. Цифровое управление ККМ, а следовательно и алгоритмы, реализованные программной частью микроконтроллера, функционально не должны уступать возможностями аналоговых микросхем ККМ с широтно-импульсной модуляцией. Кроме того, микроконтроллер или цифровой сигнальный процессор должен постоянно отслеживать состояние ККМ на отсутствие аварийного превышения электрических и тепловых параметров и уметь оперативно реагировать в случае их возникновения с возможностью индикации нештатных ситуации для визуального контроля пользователем.

Цифровой ККМ требует наличие у микроконтроллера:

- д) Аналого-цифровых преобразователей.
- е) Генератора аналоговых сигналов.

Для выполнения коррекции коэффициента мощности микроконтроллер должен иметь информацию как минимум о трех основных параметрах системы – выходное постоянное напряжение, входное переменное напряжение и ток катушки индуктивности повышающего преобразователя, являющийся основой силовой части ККМ.

Для обеспечения рабочего цикла транзистора, получения требуемой формы огибающей кривой тока может подойти программная реализация на основе ПИ-регулятора. ПИ-регулятор является хорошим выбором из-за того, что обеспечивает минимальную ошибку регулирования электрических параметров, а ширина полосы пропускания этого контура не оказывает значительного влияния на переходную характеристику системы.

Предварительно получив передаточную функцию системы по указанному контуру регулятора, можно получить форму кривой установившегося тока катушки индуктивности согласно рисунку 1:

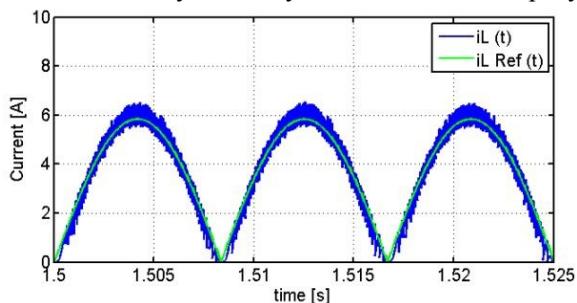


Рисунок 1 – Кривая тока катушки индуктивности

Обобщенная схема состояний ККМ (от его запуска и при непрерывном функционировании в течение времени до выключения), которые следует учитывать при разработке программы для микроконтроллера, представлены на рисунке 2:

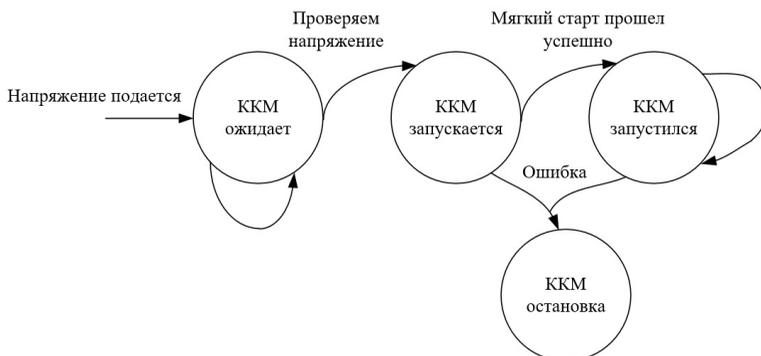


Рисунок 2 – основные состояния ККМ

Каждое из этих состояний должно быть учтено в программе, реализованной на языке высокого уровня.

Заголовочный файл должен содержать определение констант, используемых для выполнения цифровой коррекции коэффициента мощности, а также определение прототипов используемых функций. Основной файл работы должен содержать функции, предназначенные для реализации назначения устройства при расчете коэффициента мощности и всех необходимых защитных алгоритмов вплоть до полного останова и блокировки.

Для обеспечения цифрового управления следует остановить свое внимание на микроконтроллерах производителей АО «ПКК Милландр» и STMicroelectronics. Микроконтроллеры данных производителей обеспечивают требуемую производительность за приемлемую цену, частоту работы и необходимые периферийные модули, благодаря которым представляется возможным осуществлять «быстрые» виды защит КKM с реакцией за считанные микросекунды и осуществлять управление данным устройством с частотой коммутации более 100 кГц.

Заключение. Цифровые контуры управления КKM с каждым годом внедряются все сильнее в схемы импульсных источников электропитания. В этой статье был представлен один из методов программного подхода к реализации интеллектуальных КKM. Этот метод обеспечивает высокую точность и надежность получения параметров. Однако, в настоящее время до сих пор остается актуальным вопрос обеспечения более высокоскоростной реакции микроконтроллеров для обеспечения аварийных защитных функций по сравнению с аналоговыми надежными узлами и устранение так называемого «зависания», способного привести к неисправности как самого КKM и импульсного источника питания, так и подключаемой нагрузки.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Мэк, Р. Импульсные источники питания. Теоретические основы проектирования и руководство по практическому применению / Пер. С англ / Р. Мэк. – М.: Издательский дом «Додэка – XXI», 2008. - 272 с.
2. Гейтенко, Е.Н. Источники вторичного электропитания. Схемотехника и расчет. Учебное пособие / Е.Н. Гейтенко. – М.: СОЛОН-ПРЕСС, 2008. - 448 с.
3. КKM [Электронный ресурс]. Режим доступа: <https://khomovelectro.ru/korrektcii-koeffitsienta-moshchnosti.html>

УДК 621.372.542.2

В.А. ШЕВЕЛЁВРязанский государственный радиотехнический университет
имени В.Ф. Уткина**ПРИНЦИП РАБОТЫ ФИЛЬТРА НИЖНИХ ЧАСТОТ**

В данной статье рассматриваются основные принципы работы фильтра нижних частот.

Фильтр – это схема, которая предназначена для того, чтобы пропускать сигнал определённой частоты.

Фильтры имеют следующую классификацию: пассивные и активные.

Пассивные фильтры – это те фильтры, которые состоят из пассивных элементов, и они не усиливают сигнал.

Активные фильтры – это фильтры, которые помимо пассивных элементов имеют активные элементы, такие как транзисторы, операционные усилители, и они кроме фильтрации сигнала ещё его и усиливают. [1]

В данной статье разбирается принцип работы фильтра нижних частот (ФНЧ).

ФНЧ – это фильтр, который пропускает сигнал на низких частотах, и срезает, либо вообще не пропускает, сигнал на высоких частотах. Самая распространённая схема ФНЧ выглядит следующим образом:

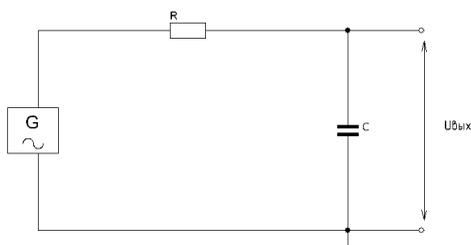


Рисунок 1 – Схема ФНЧ

Перед нами самая обычная RC – цепочка. Выходное напряжение схемы, изображённой на рисунке 1, это напряжение на конденсаторе.

Сопротивление конденсатора находится по следующей формуле (1):

$$X_C = \frac{1}{\omega C}. \quad (1)$$

Частота меняется от 0 и до $+\infty$, если вместо ω подставим 0, то X_C будет стремиться к бесконечности. А раз сопротивление большое, то и всё напряжение упадёт на нём. Если подставим вместо ω бесконечность, то X_C будет стремиться к 0, и на этом сопротивлении почти не будет падать напряжение.

Графически это можно представить такой характеристикой, как АЧХ. Амплитудно-частотная характеристика (АЧХ) – это график зависимости коэффициента усиления от частоты.

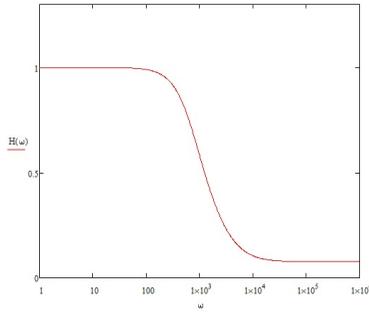


Рисунок 2 – АЧХ ФНЧ

где $H(\omega)$ – это коэффициент усиления, он также обозначается латинской буквой K . Для пассивных фильтров коэффициент усиления равен 1 или бывает даже меньше, но никогда не может быть больше единицы. Так как пассивные фильтры не усиливают сигнала, на выходе больше входного напряжения мы получить не можем. Коэффициент усиления определяется по следующей формуле (2):

$$K = \frac{U_{вых}}{U_{вх}}. \quad (2)$$

Промоделируем данную схему в Пакете Прикладных Программ ППП Micro-Cap.

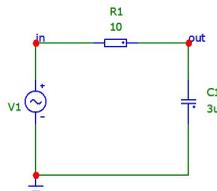


Рисунок 3 – ФНЧ в Micro-Cap

V_1 – это генератор синусоидального сигнала. In – это точка, с которой мы будем снимать осциллограмму входного сигнала, а out – точка, с которой будем снимать осциллограмму выходного сигнала. С генератора мы подаём синусоидальный сигнал, амплитудой 5 В и частотой 1 кГц. Проведём временной анализ.

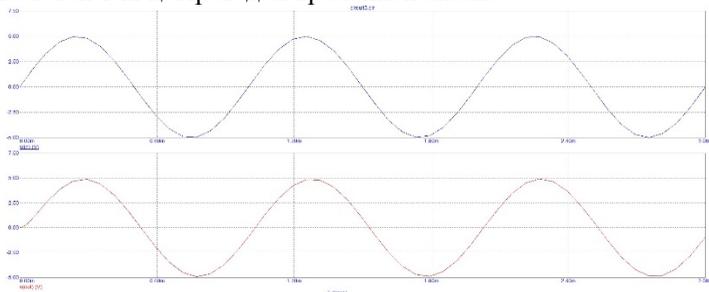


Рисунок 4 – Осциллограммы сигналов

Верхняя синусоида – это наш входной сигнал, нижняя – выходной. Как видно, что на вход мы подаём 5 В, то и получаем на выходе наши же 5 В, то есть перед нами пассивная цепь.

Теперь для того чтобы убедиться, что данная схема точно является ФНЧ и срезает высокие частоты, проведём частотный анализ.

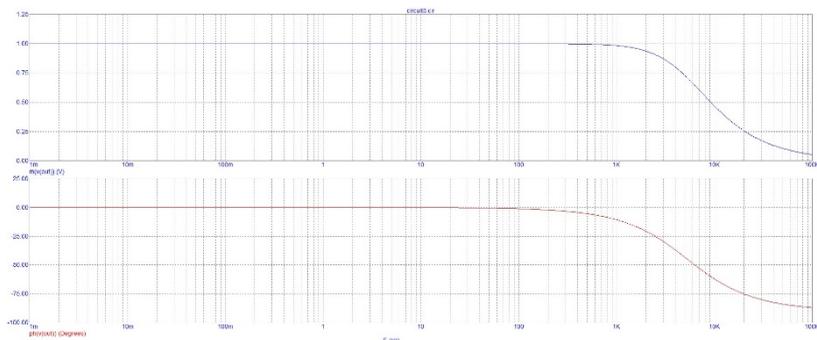


Рисунок 5 – АЧХ и ФЧХ ФНЧ

Верхний график – это и есть АЧХ, нижний – ФЧХ. ФЧХ (фазо – частотная характеристика) – показывает сдвиг фаз, угол, между сигналами, а сама по себе характеристика – это есть зависимость сдвига фаз от частоты. Как видно по АЧХ, низкие частоты схема пропускает, а высокие срезает. То есть практика полностью подтвердила теорию.

Немаловажным параметром является частота среза. Её по правилам схемотехники определяют, как коэффициент усиления, умноженный на 0.707. Выглядит это следующим образом:

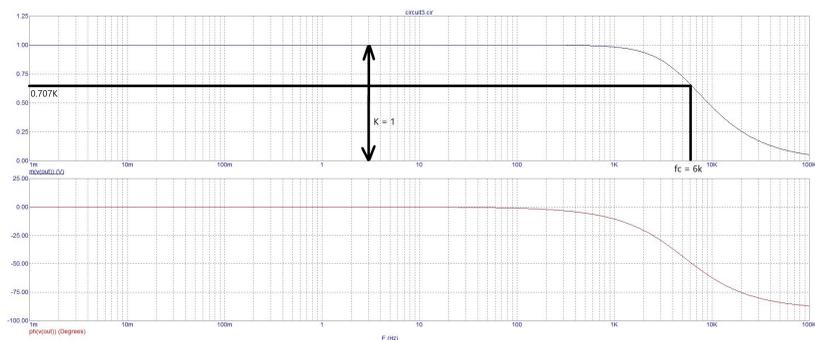


Рисунок 6 – Определение частоты среза

Частота среза равна: $f_c = 6$ кГц.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Перепелкин Д.А. Схемотехника усилительных устройств. Учебное пособие для вузов – 2-е изд., испр. и перераб. – М.: Горячая линия – Телеком, 2014. – 238с: ил.
2. Титце У., Шенк К. Полупроводниковая схемотехника: справочное руководство. Пер. с нем. – М.: Мир, 1982. – 512 с., ил.

УДК 004.03

Н.Ю. ЮРКОВРязанский государственный радиотехнический университет
имени В.Ф. Уткина**ЧАТБОТ В TELEGRAM КАК ОСНОВА ПЛАГИНА ДЛЯ LMS
MOODLE**

Рассмотрены преимущества применения чатбота в Telegram для расширения возможностей LMS Moodle. Осуществлен обзор инструментов реализации чатбота.

В современном мире дистанционное обучение становится все более значимой частью процесса обучения человека. Востребованность данного метода получения знаний обусловлена снижением материальных и временных затрат на проведение занятий, увеличением охвата аудитории учащихся и возможностью индивидуального подхода к организации обучения.

Дистанционное обучение реализуется при помощи широкого набора информационных технологий. Одним из инструментов, обеспечивающих комплексный подход к решению задач дистанционного обучения, является система дистанционного обучения (СДО). На международном уровне СДО обозначается как learning management system (LMS).

На сегодняшний день СДО используются в различных организациях. В академической среде по всему миру распространена LMS Moodle. Отличительной чертой данной системы является возможность внедрения плагинов в качестве расширения базового функционала [1].

Целью разработки плагинов для LMS Moodle является удовлетворение потребностей пользователей СДО. При большом объеме наполнения системы учебными материалами возникают потребности в удобном и быстром поиске информации, а также становится необходимым управление элементами и функциями системы через общедоступный и более распространенный пользовательский интерфейс [1].

Для удовлетворения данных потребностей целесообразно применение бота, использующего пользовательские команды для взаимодействия с элементами СДО. Бот – специальная программа, выполняющая действия по определенному алгоритму на основе указаний пользователя. Выполняемые действия носят однотипный и повторяемый характер с максимально возможной скоростью

исполнения. В настоящее время прослеживается тенденция к увеличению доли применения ботов в различных сферах экономики.

Разнообразие ботов образует множество программ, в состав которого входят чатботы. Чатбот – программа, способная отвечать на вопросы пользователя и запрашивать необходимую информацию. Чатботы находят наибольшее применение в отраслях недвижимости, образования, туризма, здравоохранения и финансов. В большинстве случаев средой для работы чатбота служит мессенджер. Одним из наиболее популярных и удобных для внедрения чатботов мессенджеров является Telegram.

Чатбот создается при помощи специального Telegram Bot API, представляющего собой HTTP-интерфейс. При создании каждому боту присваивается уникальный токен. Общие настройки бота, такие как имя бота, его описание, устанавливаются с помощью BotFather [2].

После регистрации бота необходимо запрограммировать его основные функции. Для разработки может быть использован ряд языков программирования, в число которых входит Python. Подключение библиотек PyTelegramBotAPI и BeautifulSoup4 позволит реализовать пользовательские команды.

Ответ на команды осуществляется с помощью `message_handler`, в который передаем параметр `commands`, равный массиву команд, и `send_message`, который отправляет сообщение на `id` чата. Обработка текстовых сообщений и ответ на них осуществляется с помощью передачи `message_handler` параметра `content_types=['text']`. Для ответа на картинки, документы и другой контент используются типы `photo`, `document`, `audio`, `video`. Используя указанные операторы и `register_next_step_handler()` возможно построить логическую цепочку действий чатбота [2].

Быстрый доступ к учебным материалам, контактным данным и другой полезной информации возможен за счет парсинга сайта СДО. Парсинг – автоматизированный сбор и систематизация неструктурированной информации, размещенной на веб-ресурсе. На основе функций библиотеки BeautifulSoup4 реализуется парсер страниц СДО. Оператор `urllib.request.urlopen` позволяет обратиться по заданному адресу и получить искомые данные. Оператор `urllib.error.HTTPError` выводит сообщение об ошибке выполнения поиска. Парсер возвращает данные на основе пользовательских запросов.

Взаимодействие чатбота с Telegram осуществляется с помощью отправки и получения сообщений, являющихся JSON-объектами.

Существует два метода получения сообщений от Telegram: long polling и webhook. Оба метода не могут применяться одновременно [2].

Метод long polling заключается в периодическом опросе сервера Telegram на наличие сообщений для бота. Главное отличие long polling от общего метода polling, подразумевающего мгновенный ответ на запрос, заключается в ожидании ответа на отправленный запрос в течение установленного времени. Для реализации long polling используется метод getUpdates с установленным параметром timeout, отражающим время ожидания ответа на запрос [2].

Метод webhook заключается в отправке уведомлений со стороны Telegram о сообщениях и командах для чатбота. Для получения уведомлений требуется http-сервер с выделенным ip-адресом и установленным ssl-сертификатом. После получения уведомления клиент решает, как реагировать на сообщение, а также посылает ответ «Ок, 200», означающий обработку уведомления. Для реализации webhook используется метод setWebhook [2].

Сравнение long polling и webhook и выбор между ними основываются на преимуществах и недостатках методов. Преимуществами long polling являются простая настройка метода, возможность запуска чатбота без использования обратного адреса (хостинга и доменного имени). Недостатками long polling являются большая нагрузка на сервер в связи с частой передачей неинформативных ответов на запросы, необходимость обработки массивов JSON-объектов.

К положительным сторонам webhook относится передача только информативных уведомлений при получении сообщений, адресованных чатботу, за счет чего снижается нагрузка на сеть. К недостаткам относятся более сложная настройка метода, необходимость использования хостинга с выделенным ip-адресом и установки ssl-сертификата.

Таким образом, использование чатбота в Telegram для взаимодействия с LMS Moodle позволит снизить затраты на обработку типовых запросов от обучающихся, обеспечит возможность мгновенного получения необходимой информации в любое время суток, а также увеличит уровень обслуживания пользователей. Выбор технологий, в том числе языка программирования, подключаемых библиотек, методов long polling или webhook, зависит от функционала чатбота и технических возможностей сервера.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Официальная документация LMS Moodle [Электронный ресурс]. – Режим доступа: https://docs.moodle.org/310/en/Main_page (дата обращения 15.04.2021);
2. Официальная документация для разработчиков Telegram-ботов: – Режим доступа: <https://tlgm.ru/docs/bots> (дата обращения 15.04.2021).

Информационные технологии

Межвузовский сборник научных трудов

Компьютерная верстка: А.Н. Сапрыкин,
А.С. Гавриленко, Т.С. Конкина

Подписано в печать 20.05.21. Формат бумаги 60x84 1/16.
Бумага офсетная. Печать струйная. Усл. печ. л. 12,95.
Тираж 100 экз. Заказ № 63; 4.

Издательство ИП Коняхин А.В. (Book Jet)

Отпечатано в типографии Book Jet
390005, г. Рязань, ул. Пушкина, д. 18
Сайт: <http://bookjet.ru>
Почта: info@bookjet.ru
тел.: +7 (4912) 466-151

ISBN 978-5-907400-45-0

